Column-generation and interior point methods applied to the long-term electric power-planning problem

PhD Student: Adela Pagès
Directed by: Prof. Narcís Nabona

Universitat Politècnica de Catalunya

9th November 2006

In the preparation of this thesis Adela Pagès was co-advised by Prof. J. Gondzio during two periods of study totalling 5 months at the School of Mathematics, Univ. of Edinburgh, Edinburgh, U.K., and by Prof. J. E. Mitchell during a one-month stay at Rensselaer Polytechnic Institute, Troy, NY, USA.

Adela Pagès received a Scholarship from the *Programa Nacional de Formación de Profesorado Universitario* of Ministerio de Educación y Ciencia (Spain), during the period January 2003 – December 2006.

The work in this thesis was carried out in the Research Group GNOM (Group on Numerical Optimization and Modeling) in the Department of Statistics and Operations Research, UPC, Barcelona, under Projects DPI2002-03330 (Coordinación hidrotérmica a corto y largo plazo de la generación eléctrica en un mercado competitivo) and DPI2005-09117-C02-01 (Planificación de la generación eléctrica a corto y largo plazo en un mercado liberalizado con contratos bilaterales) of Ministerio de Educación y Ciencia (Spain).

Contents

Chapter 1

Introduction

1.1 Motivation

Operations research is the use of mathematical models, statistics and optimization algorithms to aid decision-making. This thesis presents an approach to the long-term planning of power generation for a company participating in a liberalized market organized as a pool. The goal of this thesis is two-fold: to model the problem and to develop and implement appropriate and efficient techniques for solving it.

Our model gives answer to questions of the type "how much are our units expected to produce over the next year", which may arise when preparing the annual budget. The answer is not only the total generation but also the optimal distribution of the generation among the units and in the intervals into which the long-term period is split. This solution should be used in fuel procurement logistics in order to plan when to buy, for example, a ship-load of coal, or how to manage gas consumption under a take-or-pay contract.

The model must be general to allow the inclusion of any constraint regarding the availability of resources (such as water, coal or others) or any agreement settled by the company (such as take-or-pay contracts, or the minimum generation time of units).

Given that the solution of the model is a plan for several time intervals, it also provides guidelines for short-term operation so that this does not override the long-term decisions.

The long-term generation planning model is also used to simulate the effect of new generation plants by using different data to solve the problem and comparing the solutions.

It is important to note that the aim here is to find an optimal long-term generation plan for a defined situation rather than a mere simulation.

The developments proposed in this thesis were decided upon after studying and analyzing the Spanish electricity system. However, they could easily be extended to other markets structured similarly to the Spanish case.

The liberalization of the Spanish electricity market dates from January 1998. The market was structured as a pure centralized pool, in which all generation had to be bid for at the pool, the most important market being the day-ahead. The models used up to that point to plan the generation of units in a given specific generation company (SGC) had to be updated. Two main sources of uncertainty appeared as a result:

- The price paid for the electricity generation produced each hour is uncertain as it is determined by an auction in which all the units of the pool compete. All bids accepted are paid at the market clearing price.
- No SGC has a load of its own. The only load that can be predicted is the pool demand.

Moreover, in a competitive framework the goal of an SGC changes from cost minimization to profit maximization. In a perfect competitive market both goals would produce the same results [18], but this is not the case in an imperfect oligopolistic market, such as the electricity one.

This system has recently been changed (July 2006) to a mixed pool system, comprising a form of bilateral contracts in addition to the pool. The modeling and tests of this thesis are based on the pure pool system and market data from the Spanish system up to July 2006. Data for evaluating models with bilateral contracts were not available for the Spanish electricity system during the development of this thesis.

1.2 Objectives

This thesis was conducted within the GNOM *[49]* research group, which focuses on numerical optimization and the modeling of real problems – the two main lines of research explored in the thesis – and has particular experience in the field of energy generation, which is the focus of the thesis. The two main goals of this thesis, which are related to the lines of research mentioned above, are as follows:

- To propose a model for the optimization of long-term generation planning that is realistic in the context of the Spanish electricity system.
- To determine efficient procedures for solving the resulting optimization problem.

The model is tested on several representative cases of the Spanish electricity system.

1.2.1 Objectives in modeling

Our proposal is designed for an SGC with a certain level of market power (which can be defined as those companies with more than 4% of market share). An SGC that wishes to plan how much power and when to generate cannot ignore the rest of participants of the pool. It is assumed that all the market participants will follow profit-maximizing strategies.

Our proposal for planning long-term generation is to optimize the expected generation of each unit (or the merger of several units of the same type) in the power pool over each interval into which the long-term horizon is split.

The most important constraint is matching demand, since the market always clears. The Bloom and Gallant formulation [7] is used, which models the load with a load-duration curve for each interval and requires an exponential number of linear inequality constraints. This formulation accurately takes into account the outage probability and fixed maintenance periods of the units. The management of the load-matching constraints (LMCs) leads us to envisage numerous ways of solving this problem.

Other (linear) constraints included in the model are: minimum generation time (or power guarantee), limits on the availability of fuel (water, coal, gas, etc.), maximum CO_2 emission limits or the market share of the SGC. Take-or-pay constraints are also easy to include in the model.

The basic model for long-term generation planning (LTGP) maximizes the profit for all the units participating in the pool. The profit is the revenue from generating for the pool at market price (which is stochastic) minus the generation cost. The market revenue is computed through a market-price function which is estimated from historical records. This formulation is insensitive to the type of generation resulting from the optimization. The model can be extended by including the correlation between hydro generation and price, since it can be observed that the greater the hydro generation, the lower the market price.

The long term is defined as the time horizon that covers at least one year or a longer period of up to two years. This is intended to take into account the natural annual cycles of demand and climate, which also conditions the hydraulicity. The long-term period is divided into several intervals, the length of which is defined by the user.

The first interval is usually one week (or shorter if necessary), as the resulting expected generations of the units during this interval is passed to the short-term operation model, in order to account for the long-term plan results. This thesis proposes the way in which coordination between the LTGP model developed and a short-term plan should be considered and provides a model for short-term electrical power planning adapted to the LTGP proposed and which includes the long-term results.

Another decision that needs to be taken from a long-term point of view is the joint scheduling of thermal unit maintenances with the generation planning of a particular SGC.

1.2.2 Computation objectives

The long-term solutions are based on assumptions about the future that are believed to be the closest to the real situation. However, future situations (such as fuel prices) often deviate from the forecast and a new plan is therefore required, mainly to send revised information for short-term operation. Consequently, long-term planning needs to be considered before short-term planning and whenever the real situation deviates from the forecasted parameters, so the techniques implemented must be efficient so as to provide reliable solutions in a short space of time.

The proposed model for long-term generation planning with fixed maintenance periods and an endogenous objective function results in a (positive definite) quadratic programming problem, which should not prove overly difficult to resolve. However, the exponential number of LMCs poses a challenge both to the machine memory and to the solver. Two methods for handling the LMCs are proposed and compared.

A decomposition technique exploits the fact that the LMCs plus the non-negativity bounds define a convex polyhedron for each interval whose vertices are easy to find. Thus, the problem is transformed and the variables become the coefficients of a convex combination of the vertices. The transformed problem is quadratic with linear constraints, making it suitable to be solved with the Murtagh & Saunders algorithm, which gives an optimal solution. Nevertheless, a column-generation approach is used, since the number of vertices of the polyhedron is comparable to the number of LMCs. The advantage of this method is that it does not require previous computation of all of the vertices, but rather computes them as the algorithm iterates.

The application of direct methods is computationally difficult because of the exponential number of inequality LMCs. However, a reduced subset of LMCs will be active at the optimizer. A heuristic, named GP heuristic, has been devised which gives a set of LMCs including those that are active at the optimizer. It solves a sequence of quadratic problems in which the set of LMCs considered is enlarged at each iteration. The LMCs to be considered are chosen using an estimation of the potentially infeasible constraints for the current solution point. The quadratic problems are solved with an interior point method, and warm starts are employed to accelerate the solution of the successively enlarged quadratic problems.

Another objective is to determine a suitable maintenance schedule for an SGC that can be operated jointly with the generation plan. The joint model is the combination of the proposed generation planning model and a maintenance scheduling model. The decision of when a unit should be stopped for maintenance is modeled with binary variables, which makes the problem harder to solve.

1.3 Contributions

1.3.1 Main modeling contributions

▶ A quadratic long-term profit function for a generation unit participating in a liberalized pool market. It is based on a market-price regression line over the load duration computed from historical records of market clearing prices and loads and is suited for use with expected energies that match an LDC (as obtained when using the Bloom and Gallant formulation).

▶ The extension of the long-term profit function to include the influence of hydro generation on the market price, resulting in an indefinite quadratic function. It is suited for use with expected thermal and hydro energies that match an LDC.

▶ A model for maximizing the profit of all participants in a competitive power pool using the quadratic long-term profit function (with or without the influence of hydro generation) matching the pool load expressed as an LDC for each interval and using the Bloom and Gallant formulation. Other constraints on generation are also considered.

▶ A short-term planning procedure adapted to the results of the first (weekly) interval of the LTGP model developed, and employing the primal-information coordination method of imposing short-term constraints on the energy production expected in the first long-term interval.

▶ A mixed-binary joint optimization model of the LTGP and unit maintenance, maximizing the long-term profit function and matching the LDCs by employing the Bloom and Gallant formulation.

1.3.2 Main computational contributions

This thesis is mainly applied. Special emphasis has been placed on the coding of each part, either the AMPL modeling or the coding of the algorithms: in most cases a prototype in Matlab was developed and tested before C coding. However, despite the amount of time spent on coding, the results are presented in only a small number of tables.

The main computational contributions are:

▶ Fortran coding of the Dantzig-Wolfe decomposition for the LTGP model developed using the Murtagh and Saunders algorithm with column generation, applied to a quadratic problem. It consists of two phases: the first determines a feasible point for all constraints and the second improves the current point at each iteration until the optimum is reached.

▶ AMPL coding using CPLEX barrier as the solver of the LTGP model using all the LMCs and reading the right-hand sides as data.

▶ The GP heuristic for finding a reduced subset of LMCs that contains all the active LMCs at the optimizer, thus allowing the LTGP solution to be reached by solving a succession of quadratic programming problems with an increasing (though limited) number of LMCs.

▶ The AMPL coding of the GP heuristic using CPLEX barrier as the solver for the LTGP problem. In the implementation, three variants of the calculation of the right-hand sides of the LMCs were used, which produced three relatively different solution "speeds": the first computes all the LMC right-hand sides with an external routine and AMPL then reads them from a plain text file; the second lets AMPL compute the right-hand sides of the required LMCs. The fastest way of calculating the right-hand sides is to hook a routine written in C++ to AMPL.

▶ The AMPL coding of the GP heuristic using MINOS as the solver for the LTGP problem with influence of hydro generation on market prices (this problem has an indefinite quadratic objective function). To calculate the required LMC right-hand sides a routine in C++ is hooked to AMPL.

▶ The implementation of the GP heuristic in the C programming language including:

- coding of the solver, which is the infeasible primal-dual path-following algorithm for a quadratic programming problem; it employs the Mehrotra predictor-corrector with Gondzio multiple-centrality correctors and pivot regularization in the Cholesky factorization, and
- starting the interior-point method with one of the following methods:
  - a plain cold start
  - an extension of the Grothey and Gondzio warm-start technique for a quadratic problem that has new variables with respect to the previously solved problem
  - a μ-point defined as an advanced iterate which is well-centered with respect to the central path.

▶ The solution to the joint optimization of the LTGP and unit maintenance (LTGMP), which is a quadratic mixed binary problem, was attempted through branch and bound using two types of relaxation: of the binarity of the maintenance variables, and of all the LMCs except the all-one LMCs (similar to the load-balance equation) and the upper generation limits of the units. Two procedures called MS1 and MS2 were devised to find a first incumbent solution: one based on the CPLEX branch and bound solver and another following similar ideas to those of the GP heuristic. The branch and bound implemented takes advantage of the presence of special-ordered set of type 1 binary variables in the problem.

▶ The coding of the LTGMP solution approach with an AMPL script and calling CPLEX for solving the relaxation of the problems. External routines employed, also written as AMPL commands, are:

- A heuristic to find an initial feasible solution, first finding a maintenance schedule and then solving the generation planning with the GP heuristic. Two heuristics for determining an incumbent maintenance schedule arise:
  - The MS1 heuristic, which relaxes only the LMCs and solves the mixed integer problem with CPLEX to optimality or near optimality.
  - The MS2 heuristic, which builds the maintenance schedule by ordering the units which are more likely to be in service. Like the GP heuristic, it solves a sequence of quadratic problems in which some LMCs are appended and some units are fixed to be in service.
- The GP heuristic, through which the actual objective function of a solution with fixed maintenance periods is computed.

▶ The preparation of all the test cases described in the appendix.

▶ The solution, with appropriate test cases, of all computational procedures developed.

▶ Comparison of the performance of:

- an existing code implementing the active set method with row generation and the Dantzig-Wolfe decomposition procedure developed.
- two variants of the Dantzig-Wolfe decomposition procedure with column generation.
- the GP heuristic for the LTGP problem with the three types of coding of the calculation of LMC right-hand sides.
- the different warm-start procedures applied in the interior point solutions of the quadratic programming problems solved in the GP heuristic.
- the GP heuristic implemented in AMPL or C-coded with an interior point code.
- the different procedures employed to solve the LTGMP problem, the joint optimization of the LTGP model and the units' maintenance.

1.4 Contents

The chapters in this thesis are:

1 Introduction

The first chapter outlines the motivations behind the study, its objectives, contributions and a summary of the contents.

2 The long-term generation planning problem

This chapter defines the Spanish electricity system in the liberalized age (from 1998 to 2006) and presents the basic modeling concepts of long-term power generation planning, its extension taking into account the influence of hydro generation and the extension to solve the joint model of generation planning and the maintenance schedule.

Basic modeling concepts are: the load-duration curve that models the demand, the generation-duration curve, which is the expected generation that matches the load-duration curve, the loading order with which the expected generations are computed, the load-matching constraints and the long-term market-price function from which the expected revenue is computed.

3 Column-generation methods

Chapter 3 explains the details of the transformation of the long-term generation problem, how to find the vertices and the steps of the Murtagh & Saunders algorithm.

Like any active set method, the Murtagh & Saunders algorithm prices the active constraints at each point, both the general constraints and the non-negativity active bounds. When an active bound happens to have a negative multiplier, the vertex associated with it is computed and included to check whether it improves the current objective function or not. Two variants of the algorithm are tested: one that stores all the vertices computed and another that discards the nonbasic vertices. An analysis of the algorithm performance and a comparison of both versions is carried out in several realistic test cases from the Spanish power pool.

4 Direct solution methods and heuristic procedure

This chapter presents the heuristic developed for finding a generation plan, the GP heuristic. The solution given by the heuristic does not have to be feasible or optimal. This has to be checked "by hand" (looking at all possible LMCs). Certain situations are described in which we can ensure that some LMCs do not have to be checked because for the solution found they must be feasible.

Computational results are given for the solutions obtained with the heuristic and these are then compared with the column-generation solution.

The last section is devoted to showing the heuristic for a small test case.

5 Application of interior point methods

The heuristic proposed in chapter 4 (the GP heuristic) solves a sequence of quadratic problems. These problems are solved with a primal-dual path-following method.

This chapter outlines the steps of the interior-point method employed, with some implementation details such as the search direction: the Mehrotra predictor-corrector plus multiple centrality correctors, or the way the Newton system is solved.

The computational results section compares the performance of the multiple-centrality corrections and the solutions obtained with our own code or the AMPL implementation using the CPLEX barrier solver.

6 Warmstarting for interior point methods

Warmstarting in interior point methods refers to the techniques that exploit any information about the problem (such as the solution of a related problem) to give an initial point for the usual algorithm that should approach the optimizer faster than starting the algorithm from scratch (called cold start).

This chapter extends the warm start proposal of Gondzio and Grothey to the quadratic programming case for initializing a sequence of problems of differing size.

Standard practice is to store an advanced iterate of the previous solution. From this point, some Newton-type steps are performed in order to recover primal and dual feasibilities while not altering too much the complementary products.

Warm-start techniques are applied to the sequence of problems given by the GP heuristic. Efficiency is compared for both computation time and iteration savings.

7 Long- short-term coordination

This chapter deals with how long-term planning creates a framework for short-term planning. This is the factor behind the research done in order to find reliable solutions in a short time.

From a practical point of view, the long-term generation planning model must be solved regularly prior to any short-term planning solution, for budgeting, and whenever the forecasted situation varies, in order to pass sound information to the short-term plan.

This chapter contains the development of a model for short-term planning. The main characteristic of this model is the procedure proposed for building the market price. It is built on estimations of the (supply/demand) bidding curves. In this model the demand is assumed to be inelastic and two different supply curves are employed: a linear curve used to determine the unit commitment and the zero-priced generation bids, and a nonlinear one using a polynomial of degree 4 with a linear term dependent on the generation level to obtain the final generations and market prices through nonlinear programming, using the unit commitment results and zero-priced bids found previously.

8 Solution approach for the long-term problem with maintenances

The joint model for the long-term generation planning and maintenance scheduling is a quadratic mixed binary problem. The optimal solution for such a problem is hard to determine.

In this chapter two heuristics are proposed, MS1 and MS2, which represent two alternatives for establishing a good maintenance schedule. The maintenances are then fixed and the final generation plan is found with the GP heuristic.

An alternative approach is to solve the problem with a branch-and-bound procedure. This chapter contains the details of the branching and bounding rules used in our AMPL implementation of the algorithm.

The performance of the procedures proposed and the CPLEX solution is discussed in the last part of the chapter.

9 Conclusions of the thesis

The final part of the thesis contains the main results obtained, suggestions for further research and publications and presentations generated by this thesis.

Chapter 2

The long-term generation planning problem

This chapter presents the models that are used in the following chapters. It starts with a description of the Spanish electricity system from the standpoint of the electricity market. It follows with the basic modeling concepts such as: the load-duration curve, which models the demand, the generation-duration curve, which is the expected generation that matches the load-duration curve, the loading order with which the expected generations are computed, the load-matching constraints and the long-term market-price function from which the expected revenue is computed.

The matching of the load is formulated with the Bloom and Gallant model, which uses an exponential number of linear inequality constraints. There is a section devoted to show the relationship between the LMCs and the loading order of the units. The rest of constraints used for modeling the electricity system for a long-term planning (e.g., minimum service time or the SGC market share) are expressed as linear constraints. The emission allowance constraints are incorporated into the objective function, given that the emission allowances can be traded in the ETS market. The condition that the load should be matched at all times becomes a very relevant characteristic of our model. In the following chapters two solution methods are compared in order to avoid the computation of all the LMCs.

The market-price function with respect to the load duration is extended to take into account the hydro generation level, given that in the long term a negative correlation between hydro generation and price is observed.

The last section limits the scope of this thesis regarding the modeling of the stochasticity and market equilibrium.

2.1 The Spanish electricity market evolution

2.1.1 Electricity cycle

Electricity is a unique commodity because it cannot be stored and has high technical requirements. The electricity cycle is composed of four main stages: generation, transportation, distribution and consumption. Transportation refers to the transfer of electricity through the high voltage network and distribution is bringing the electricity from the high voltage network to the consumption points.

2.1.2 Regulated and liberalized electricity systems in Spain

Historically, the supply of electricity was carried out in Spain by private owned vertically integrated utilities. Generation, transportation and distribution were managed by the same utility over a certain area. Electricity prices were fixed by the government.

In 1998, Spain started the liberalization of its electricity market. The reason for liberalization is to introduce competition by creating a power exchange market, which should reduce costs. Some years before, the transportation network had been acquired by a publicly regulated company.

With the liberalization of the electricity market, the companies, each of which had previously held a monopoly in its area, had to split into separate businesses: generation, distribution and commercialization. As a result, an area no longer belongs to a single company; instead, many national and foreign companies may operate within it. Moreover, because of the creation of new businesses, other participants move into the electricity market, making the system more competitive.

The pioneers in this movement were Norway, Sweden, Finland and Denmark, which joined their electricity systems in 1993 and created the Nordic Power Exchange, known as Nord Pool. Shortly thereafter, the United Kingdom started to liberalize its electricity system with the New Electricity Trading Arrangements (NETA).

The liberalization of electricity is a long process due to the complexity and importance of electricity generation. Perfect competitiveness is impossible; by nature, the generation of large amounts of electricity is an oligopoly (there is an important cost barrier). Moreover, market rules cannot override the essential electricity service, so market regulation has to guarantee supply and its quality. There is also an intrinsic complexity in power generation in that the system has to be balanced at all times.

Liberalization adds two new activities to the electricity system: the wholesale trading of power generation and retail commercialization to the end consumers. The first stage of liberalization is the wholesale trading. Two new entities are created: the system operator (SO), which ensures the reliability and quality of the system and controls the transportation network, and the market operator (MO), which regulates a number of electricity markets and ancillary services.

The Spanish electricity system offers two possibilities for electricity trading:

- in an organized power-exchange market, also called Pool,
- or through a bilateral contract, which is an agreement between the supplier and the consumer for a fixed quantity at a specific negotiated price for a given future period.

The main electricity market is the daily spot market, which is complemented by minor markets of ancillary services and six intra-day markets.

Market participants are companies authorized to participate in the electricity production market as buyers or sellers. Sale and purchase bids are presented to the MO which matches the lowest bids. The last bid matched gives the marginal energy price. The accepted generation bids plus the bilateral contracts are passed to the SO which checks the physical feasibility.

In Spain, the spot market is mandatory: all available units must make a sale bid or have a bilateral contract.

One difference between the trading systems is that in the organized market the bids stand for a maximum of one hour whereas a bilateral contract is an agreement for a longer period, usually for one year. However, additional regulations, which were in force until July 2006, in practice discouraged bilateral contracts, as a unit participating in a bilateral contract (even partially) was not entitled to claim a power guarantee bonus. This bonus is given to units that generate for the pool for the equivalent of at least 480 hours per year at full capacity. Therefore, it can be said that the Spanish electricity system was a *centralized pure pool system* up to July 2006.

The second stage of liberalization is the retail trading. In Spain, since 2003 any consumer is allowed to choose its supplier and negotiate the electricity price or stay with the regulated prices. In practice, there are resellers who sell the energy to consumers and buy the energy from the market or through a bilateral agreement.

2.1.3 The MIBEL

The Iberian Electricity Market (MIBEL) is the union of the Spanish and Portuguese electricity markets. This is a natural extension due to the geographical situation of the two countries. The new market, launched in July 2006, is larger and therefore increases competition between its participants.

The MIBEL organizes electricity Futures markets. A Futures market is an auction market in which participants buy and sell energy contracts for delivery on a specified future date. The transactions can be settled both by physical delivery of by differences with respect to the spot price. Futures markets allow the market participants to hedge their long-term costs.

During the initial phase, the Portuguese MO (OMIP) governs the Futures market and the Spanish MO (OMIE) rules the daily market. In the future, OMIP and OMIE must merge to create a single market operator: the Iberian MO (OMI).

The SO of each country remains responsible for the complying with the technical constraints.

2.1.4 The emission allowance trading system

European Union (EU) member states are committed to achieving the objectives set out in the Kyoto Protocol. The Kyoto Protocol is aimed at lowering overall emission of six greenhouse gases in order to prevent major climate changes due to the global warming.

Each country that ratified the Kyoto Protocol agreed to limit greenhouse gas emissions to a certain level (6-8% from 1990 levels, in the EU). The European Union has created a system of emission trading in an effort to meet the Kyoto targets. Initially (from 2005 to 2007), the EU Emission Trading Scheme (ETS) covers only carbon dioxide (CO_2) emissions from large emitters in the power plant and heat generation industry and energy-intensive industrial sectors.

The traded commodity is the *emission allowance*, which represents the right to emit one ton of CO_2. Each EU member state allots the allowances free of charge among its CO_2-producing installations. Companies that keep their emissions below the cap or level established by their allowances are able to sell their surplus to the market and companies must buy allowances for excess CO_2 emissions.

2.2 Basic modeling concepts

2.2.1 The long-term horizon

The generation planning horizon considered is of one or two years. Such study horizon is termed *medium term* by many authors, who reserve the *long term* to problems with horizons of two to more years where generation expansion (and old unit shut-down) is considered. Generation expansion is beyond the scope of this thesis, hence, all generator units in the pool are assumed to have a preestablished availability within the horizon considered.

The long-term horizon is split into several intervals. There are two main reasons for this division: some of the parameters change over time and some constraints may apply to periods shorter than the whole horizon.

Given that one goal of the LTGP is to set up a framework for short-term scheduling, the first interval is always one week and the optimized variables for this week will be used in short-term planning.

Superscript ^{i} indicates the interval and n_i is the number of intervals.

2.2.2 Generation units

The set of generation units from the pool is Ω. Some of these units belong to the SGC. Subscript $_j$ indicates the unit and the parameter n_u is the number of units. The main characteristics of the units are ($j \in \Omega$):

- c_j: capacity (in MW)

  It is assumed that a unit will generate as much as it can. This is a reasonable long-term approach.

- q_j: probability of random failure

  A unit may not be available when it is required to generate. The complementarity is the in-service probability: $p_j = 1 - q_j$.

- f_j: linear generation cost (€/MWh)

- $\mathcal{M}_j$: set of intervals in which unit j is not available because of maintenance (when this is a fixed decision).

The set of available units may change for some intervals due to fixed maintenance planning. Ω^i shall be taken as the set of units available in interval i, and n_u^i the cardinality of this set:

$$\Omega^i := \{j \in \Omega \,|\, i \notin \mathcal{M}_j\} \quad \forall\, i$$

2.2.3 The load-duration curve

The load-duration curve (LDC) is the most sensitive technique for representing the load of a future interval i. The main features of an LDC are:

- t^i: the duration,
- $\overline{p}^{\,i}$: the peak load,
- $\underline{p}^{\,i}$: the base load,
- $\widehat{e}^{\,i}$: the total energy, and
- the shape, which is not a single parameter but is usually described using a table of durations and powers.

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 the LDC maintains the power variability of the load in its entirety.

LDCs provide a rational way of considering interval loads in LTGP. There are two different methods for considering LDCs:

- One in which the LDC is split into several time blocks with uniform power. The number of blocks may vary from only a few (e.g. for the peak, for the mean and for the base hours) to one for each hour of the interval. LDCs modeled in this way will be termed *block-LDCs*.
- Another in which the LDC is considered to be a continuous function of load duration or, equivalently, where the load duration, or the probability of not having less than a certain load, is considered a function of the power; this latter case corresponds to the load-survival function. This will be referred to as the classical LDC, or simply the LDC.

It is common in the literature on LTGP employing the block-LDC [4] for each block to be treated as a short-term load and for generation units to match this load by simply balancing its power. When using the classical LDC the load is matched taking into account outage probabilities and calculations follow the convolution rules established in [2]. The unit energies obtained using the convolution procedures are true expected values, whereas those obtained with the former procedure are not.

2.2.4 Expected generation given a loading order

For the sake of notation simplicity, this section omits the index ^{i} of the interval, however the expected generation and loading order always refer to a particular interval.

Given the set of all generator units participating in the pool, we wish to match the demand of the whole pool, which can be considered as a random variable. Let x_j be the expected energy generated by unit j over an interval of length t. For a given density function of the demand, $f(z)$, it is computed as:

$$x_j = t(1-q_j)\int_0^{c_j} zf(z)\,dz = t(1-q_j)\int_0^{c_j}[1-F(z)]\,dz$$

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

$S(z) = 1 - F(z)$ is called the load-survival function and gives the probability of a load greater than z, $S(z) = prob(\text{demand} \geq z)$. We thus have:

$$x_j = t(1-q_j)\int_0^{c_j} S(z)dz\,.$$

Either density or survival functions can be used interchangeably, since one can be derived from the other. Here the survival function is preferred, as it corresponds to the LDC of the whole pool rotated and rescaled, which can be predicted with acceptable accuracy.

The load is matched with the available units. The expected generation of each unit depends on the loading order. Let ω be a subset of units ordered before unit j. The expected contribution of unit j ordered after units in ω is:

$$x_{j|\omega} = t(1-q_j)\int_0^{c_j} S_\omega(z)\,dz \tag{2.1}$$

Function $S_\omega(z)$ is the load-survival function of still-unsupplied load after loading units in ω. Balériaux et al. [2, 68, 8] were the first to propose the following convolution:

$$S_{\omega\cup j}(z) = q_j S_\omega(z) + (1-q_j)S_\omega(z+c_j)\,, \tag{2.2}$$

which expresses the change to the load survival function caused by loading unit j.

Let $S_\varnothing(z)$ be the load-survival function corresponding to the LDC (prior to loading a unit). Thus, by successively applying (2.2) it can easily be deduced that – once all the units in ω have been loaded – the unsupplied load will have the following survival function:

$$S_\omega(z) = S_\varnothing(z)\prod_{m\in\omega} q_m + \sum_{\chi\subseteq\omega}\big(S_\varnothing(z+\sum_{j\in\chi} c_j)\prod_{j\in\chi}(1-q_j)\prod_{j\in\omega\setminus\chi} q_j\big) \tag{2.3}$$

where χ represents any subset of ω.

It can thus be stated from (2.3) that the survival function $S_\omega(z)$ of the unsupplied load is the same regardless of the order in which the units in ω have been loaded.

It is easy to verify from (2.2) that $S_{\omega \cup j}(z) \leq S_\omega(z),\ \forall\, z$. It then follows from (2.1) that loading a unit k after loading unit j will yield a lower expected production x_k than if unit k were loaded just before unit j.

The expected unsupplied energy after loading units in ω, $w(\omega)$, is computed as follows:

$$w(\omega) = t \int_0^{\overline{p}} S_\omega(z)\, dz \tag{2.4}$$

The integration in (2.4) is to be carried out numerically.

2.3 State of the Art in LTGP

Before the advent of the electricity markets, a company did its annual planning by matching a single load-duration curve that corresponded to its area of influence for a period of one year. In 1967 Baleriaux *et al.* [2] proposed the convolution method as a tool for computing the expected generation of each unit given a loading order for matching an LDC. Such a model is known as a *probabilistic production costing model.* For many years, the loading order (based on the cost merit order) was built heuristically.

In 1994 Bloom and Gallant [7] formulated the calculation of the expected generations that match an LDC subject to resource-limiting constraints as a linear optimization problem that finds the optimal loading order for a single interval. This model, although theoretically very elegant, is impracticable as regards its direct application to real-size problems as it employs an exponential number of inequality constraints. The same authors proposed an active set methodology for their solution, which does not explicitly require all the constraints.

In 2000 Pérez-Ruiz and Conejo [53] presented an extension of the Bloom and Gallant model, with the long-term period (usually one year) divided into shorter intervals and solved through a column generation method. This multi-interval approach allows a more accurate representation of time-dependant constraints (such as the availability of water in some months). In Nabona *et al.* [44], there is a detailed description of the implementation of the active-set method to solve the Bloom and Gallant formulation of the long-term multi-interval planning with several types of operational constraints.

Little research has been carried out to bring the production costing models up to date so that they include the competition among generation companies. This thesis extends the Bloom and Gallant model by including the market.

From a long-term point of view, the electricity market system [58] has been modeled for several purposes, which can be grouped into two main trends: the analysis of market behavior and the planning of the generation over such a period. Since the deregulation of electricity a diversity of models have been proposed for analyzing the market. Ventosa *et al.* [66] present an extensive survey regarding modeling of the electricity market.

The models that simulate the electricity market need to assume some strategy of its participants. A first categorization divides companies into price makers, which are able to alter the market price by changing its generation, and price takers: companies whose behavior does not influence the market price. Then a strategy profile has to be defined for the price maker companies. It is assumed, given the participants' strategies, that the market price is a Nash equilibrium, that is, the solution in which none of the participants are motivated to change their strategy as this will not increase their profits. The most popular equilibrium model for the electricity market is the Nash-Cournot equilibrium. The Cournot oligopoly model assumes that the participants compete in quantities and that competitors do not respond to price changes. The price is derived from the *inverse market demand*. Another special model is the Bertrand one. It assumes that the participants compete in price, and this leads to a solution where the marginal revenue equals the marginal cost. This is a situation of pure competition. Day *et al.* [18] present an extensive survey on possible participant strategies and the resulting equilibrium of the models (if they exist).

Liu *et al.* [36] address the analysis of the market by formulating it as an optimal control problem, which simulates the operation of the market. The main result of these models is a prediction of the market price, to be either employed for assessing the profitability of a future contract, or for showing that market prices may increase if price-maker companies exercise their market power.

The models that simulate the market find a forecasted market price as a solution of the model, and sometimes the expected generations can be deduced from their results [4, 63]. Nevertheless, when historical data is available, statistical techniques for modeling, estimating and forecasting the market prices can be used. This is the case in [39], where Markov chains have been used considering the set of scenarios modeled as a Markov model with several states. Statistical techniques with historical data have been also used in this thesis to find through regression a market-price function with respect to load duration, and the change of this function with hydro generation.

One of the main effects of electricity system deregulation is the uncertainty of the market-price. Bilateral contracts or trading in the Futures markets (provided that the Futures market has been organized) become a tool to hedge against price variation in the long term. Kelman *et al.* [34] study the market power in hydrothermal systems assuming that a Nash-Cournot equilibrium with an inelastic demand is reached by the players. They show the differences in the spot price between a Nash-Cournot or a least-cost strategy and that the market power diminishes with the number of price-maker generators. Bilateral contracts are a tool for reducing market power.

There are other lines of research which sit in-between the long- and the short-term planning, such as in [36] where a procedure for finding short-term bidding strategies which are based on an analysis of the market in the long-term is proposed.

2.3.1 Other long-term generation planning approaches

Several authors have proposed alternative models for long-term generation planning in a competitive market.

Bjørkvoll *et al.* [55] approach generation planning and risk management as two-phase decisions, where the risk management model includes the optimal generation planning. The SGC considered is a price-taker company. Given that their model does not include constraints that link the generation of different units, the model is decomposed by units, and the maximization of the profit is done independently for each unit. Uncertain parameters, such as the price, are taken into account via a scenario tree. Then a maximization of the profit is done independently for each unit, interval and scenario. The minimization of the risk is done taking into account bilateral/futures contracts made by the SGC. The objective function is the maximization of the overall expected profit, which is the profit from the units owned (computed from the hydro and the thermal scheduling step), plus the profit from trading in the futures market, minus the expected costs of a shortfall.

Kasenbrink *et al.* [35] propose an integrated model for the annual generation planning and the trading of generation, both in the day-ahead market and in the Futures one. The model assesses the risk of taking a decision with some uncertain variables and includes a risk aversion parameter that is fixed by the user, where the risk depends on the position of the company, that is the balance between sales and supplies. The market revenue (or costs, if the company buys more than it sells) is expressed quadratically with parameters that capture the market reaction with respect to the behavior of the considered company. The year is split into monthly intervals but no multi-interval constraints (such as fuel limits) are considered. They use stochastic programming with scenarios to represent the stochasticity of some parameters, and propose a Lagrangian relaxation to solve their formulation.

Another approach proposed by other authors for planning long-term generation focuses on the procedure of computing a market equilibrium. For example, Barquín *et al.* [3] first formulate the equations of an equilibrium point and then use a minimization problem as an equivalent formulation. This optimization problem is enlarged with technical constraints, and given that some parameters are uncertain, the market equilibrium is solved with stochastic programming techniques. The market equilibrium is represented by means of a *conjectural variation*, where each company's generation bids are linear functions of the price, as is the market demand. The market price is derived from the solution of the market equilibrium giving a single price for each period considered.

Cabero *et al.* [9] present a three-step procedure for solving a formulation of the integrated risk-management problem. The risk can be managed either by participating in the Futures market (of electricity, fuel or currency) or by changing the company's operational decisions (by shifting periods of hydro generation). The results given by the model support both the operational and the financial decisions. First of all, several scenarios are defined (in terms of water inflows, demand and fuel prices). For each scenario a equilibrium-market model for the long-term operation is solved, which gives electricity spot prices and generation profiles for each company. Then a multistage scenario tree is defined, with which a portfolio optimization, which includes the participation in the Futures market, is done. The risk is measured with the Conditional Value-at-Risk (CVaR) value. Regarding the modeling of the load for the market-equilibrium solution, they split each period into load levels (as a block LDC), and a spot price for each load level is found as a Nash-Cournot equilibrium.

The case of a price-taker hydro generation producer in a market with bilateral contracts is analyzed by Shrestha *et al.* [60]. Uncertainties considered are the market price, which is exogenous to the company's production, and the inflows. Both these uncertainties are represented through scenarios.

Stochastic programming with recourse is employed to maximize the profit subject to water balance constraints and operational limits, the recourse actions being extra hydro generation, spillage and storage. The profit maximized is then composed of the expected profit calculated at expected prices plus a term with the expected recourse generation priced at each scenario price. The risk-averse case is also optimized by minimizing the expected squares of differences between the profit for each scenario and the expected profit found in the risk-neutral solution. This model is highly dependent on the water available because, as a price-taker producer, the company will generate as much as it can (without having to match the load).

2.3.2 Market-price analysis

Batlle and Barquín [4] present a procedure for the long-term cost and income analysis. It employs a supply-bid function for an hourly interval for an SGC in an oligopolistic market. The profit maximization conditions provide a *strategic* supply-bid function where the *market power* of the SGC is taken into account. Only hydro generation is considered to produce a zero-priced bid. Its distribution among the hours is obtained by considering the block-LDC and a given hydro-energy allotted to the interval placed in a loading-order position that matches its quantity, this being the only constraint considered. By aggregating the strategic supply-bid functions of all generation companies participating in the market, a total strategic supply-bid function is obtained for the hour, and by cutting this function with the power of the corresponding block, an estimate of the hourly market-clearing price is obtained. The estimates of the market prices of the successive blocks of the LDC produce a *market-price duration curve* of the interval, with which a long-term cost analysis can be carried out.

Valenzuela and Mazumdar [63] develop a procedure for determining the price-duration curve for a sample day (or T-hour time lapse) in a future period in an oligopolistic energy market under the Bertrand model of perfect competition and under a supply function equilibrium model described in [57], where the competing firms are identical and the load has zero elasticity. Hourly loads are considered to be normally distributed random variables with known mean and standard deviation, and it is assumed that all the generators in the market will be loaded in merit order, taking into account their forced outage rates. The formula for the hourly market price in the symmetric oligopoly presented in [57] is extended to consider generator random outages. Under both models, the load-survival function after loading units in merit order is developed and computed using the approximation of the cumulants method. The price-duration curve follows easily from the load survival function by using the average of the market prices for each of the T hours considered. Computational results show a reasonable agreement with real price-duration curves, given that real markets are not symmetric oligopolies.

The approach presented by Valenzuela *et al.* [63] is interesting from the standpoint of our model. However, the hypothesis of the symmetry of the companies is unrealistic and the assumption that the units are sorted in cost merit order does not hold when there are fuel limits and other operational constraints.

2.4 The Bloom and Gallant formulation

In this section the interval index ^{i} has been omitted, but the results and notation should be read as they refer to a particular interval.

In 1994 Bloom and Gallant [7] introduced a formulation for matching an LDC based on linear inequalities. The constraints are derived from the following simple observation:

> The total generation of a group of units in any loading order cannot exceed their total generation in a loading order where they are loaded first.
>
> *Bloom & Gallant, 1994*

The constraints that express this idea are called load-matching constraints (LMCs):

$$\sum_{j \in \omega} x_j \leq \widehat{e} - w(\omega) \quad \forall \, \omega \subseteq \Omega. \tag{2.5}$$

The right-hand side (rhs) of the LMCs (2.5) is the difference between the total energy demand on the system and the unserved energy after loading the units of the associated subset. From equations (2.3) and (2.4) it is deduced that the rhs is independent of the order used to compute it. This formulation takes into account any possible loading order.

Note that variable x in the load-matching constraints (2.5) is not conditioned to an ordering, although one always exists. We are interested in the expected generation rather than the ordering of the units, although both provide equivalent information.

The major inconvenience of this formulation is that there are $2^{n_u} - 1$ possible groups of units, or linear inequality constraints. On the other hand, it accurately models the probability of random failures, thus avoiding the need for scenarios to model this stochasticity.

Matrix $B^i \in \mathbb{R}^{(2^{n_u}-1) \times n_u}$ denotes the variable coefficients of the LMCs of interval i, which are 0 or 1, and vector $r^i \in \mathbb{R}^{(2^{n_u}-1)}$ is the LMC rhs. A single LMC is written as $B^i_\omega x \leq r^i_\omega$, where ω is the set of units that defines this constraint. The matrix and rhs vector with all the intervals is $B \in \mathbb{R}^{(2^{n_u}-1) \cdot n_i \times n_u \cdot n_i}$ and $r \in \mathbb{R}^{(2^{n_u}-1) \cdot n_i}$.

2.4.1 Loading order

When the units are ordered by increasing production cost, the loading order is called *merit order*. It is easy to show (see [43]) that the merit order is optimal when the objective function is cost minimization and only load-matching constraints are present.

However, this is not the usual situation. When there are other constraints the optimal generation of a unit may have a position that "*splits*" the position of another unit. Fortunately, it is not necessary to calculate the exact location of the split. This simplification arises from an important property of the model: the *energy invariance*. When one unit splits another, the total energy generated by the pair does not depend on the position of the split. This property generalizes to a situation in which many units split one another and also applies when generator outages are considered [7].

The ordering of the units is denoted by the ordered set Ω_o, which contains the indices of the units ordered. When no unit is split, we say that there is a *perfect ordering*. When there are one or more split units, we say that there is a *partial ordering*. In this case there is no order among the split and the splitting unit(s). The set of indices of units ordered before unit j is denoted by Ω_{o_j}.

Example

For example, if $n_u = 5$ and $\Omega_o = \{3, (5, 2, 1), 4\}$, it means that unit 3 is the first in the loading order and units 1, 2 and 5 precede unit 4, which is the last one, but there is no ordering among units 1, 2 and 5.

The subset Ω_{o_5} refers to unit indices $\{3, 5, 2\}$.

Note that an arbitrary ordering is fixed for the units partially ordered.

An alternative way of computing the rhs of an LMC (2.5) is:

$$r_\omega = \sum_{j \in \omega} x_{j|\Omega_{o_j}} \tag{2.6}$$

where $x_{j|\Omega_{o_j}}$ is computed as in (2.1), and the ordering of the units in ω is arbitrary for the rhs computation.

Example

Consider a case with 5 units ordered as $\Omega_o = \{3, 5, 2, 1, 4\}$. The right-hand side of constraint $\omega = \{3, 5, 2\}$ is computed as:

$$\begin{aligned} r_\omega = \; & t[(1 - q_3) \textstyle\int_0^{c_3} S_\varnothing(z)\, dz \; + \\ & (1 - q_5) \textstyle\int_0^{c_5} S_{\Omega_{o_5}}(z)\, dz \; + \\ & (1 - q_2) \textstyle\int_0^{c_2} S_{\Omega_{o_2}}(z)\, dz], \end{aligned}$$

and is valid for any other ordering in which the units in ω are the first.

2.4.2 Relationship between loading order and active load-matching constraints

The matching of a load-duration curve is carried out following a unit loading order. Each unit (or subset of units in a partial ordering) loaded implies a new active constraint, which is defined by the units already loaded plus the new one(s). Therefore, a loading order is modeled with, at most, as many active constraints as units to be loaded.

Example

Imagine a solution for a case with 5 units with loading order $\Omega_o = \{3, 5, 2, 1, 4\}$. The matrix of active LMCs is:

$$\begin{bmatrix} B_{\{3\}} \\ B_{\{3,5\}} \\ B_{\{2,3,5\}} \\ B_{\{1,2,3,5\}} \\ B_{\Omega} \end{bmatrix} x = \begin{bmatrix} . & . & 1 & . & . \\ . & . & 1 & . & 1 \\ . & 1 & 1 & . & 1 \\ 1 & 1 & 1 & . & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} r_{\{3\}} \\ r_{\{3,5\}} \\ r_{\{2,3,5\}} \\ r_{\{1,2,3,5\}} \\ r_{\Omega} \end{bmatrix}$$

If the ordering is $\Omega_o = \{3, (5, 2, 1), 4\}$ only $B_{\{3\}}x \leq r_{\{3\}}$, $B_{\{1,2,3,5\}}x \leq r_{\{1,2,3,5\}}$ and $B_{\Omega}x \leq r_{\Omega}$ will be active.

We say that LMC $B_{\zeta}x \leq r_{\zeta}$ is *nested* into $B_{\theta}x \leq r_{\theta}$, where ζ and θ are any set of units, if $\zeta \subset \theta$. In general, a set of constraints is nested if the constraints in the set can be ordered in such a way that every constraint is nested in the next.

Given that any feasible solution follows a loading order, the only LMCs that may be active are nested.

Example

Following the example, a set of nested constraints is:

$$\begin{bmatrix} B_{\{3\}} \\ B_{\{1,2,3,5\}} \\ B_{\Omega} \end{bmatrix} x = \begin{bmatrix} . & . & 1 & . & . \\ 1 & 1 & 1 & . & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} r_{\{3\}} \\ r_{\{1,2,3,5\}} \\ r_{\Omega} \end{bmatrix}$$

where $B_{\{3\}}x = r_{\{3\}}$ is nested in $B_{\{1,2,3,5\}}x = r_{\{1,2,3,5\}}$ and $B_{\Omega}x = r_{\Omega}$, $B_{\{1,2,3,5\}}x = r_{\{1,2,3,5\}}$ is only nested in $B_{\Omega}x = r_{\Omega}$ and $B_{\Omega}x = r_{\Omega}$ nests $B_{\{3\}}x = r_{\{3\}}$ and $B_{\{1,2,3,5\}}x = r_{\{1,2,3,5\}}$.

Special load-matching constraints are:

- $B_{\Omega}x \leq r_{\Omega}$ All-one LMC.

  Since an LMC by itself does not imply any ordering, the all-one LMC is valid for any ordering. This LMC nests any other constraint.

- $B_{\{j\}}x \leq r_{\{j\}}$ Upper bound of unit j ($\overline{x}_j := r_{\{j\}}$).

  Given that the capacity of a unit is generally lower than the LDC base load, the upper limit is $c_j(1 - q_j)t$, which is the expected amount of electricity produced by the unit generating all the time.

2.5 The non-load-matching constraints

There are five major groups of generation units defined according to the fuel they use for producing electricity: hydro generation units (from stored water), thermal power plants which burn either coal, oil or gas, or a mixture of oil and gas, combined cycle plants (a gas turbine and a steam turbine), nuclear plants and generation from other renewable sources (wind-power, run-of-the-river hydro and others). Some of these units require special modeling. The constraints that model the electrical system are:

- Maximum hydro generation

  Hydro generation is one of the most favored means of producing electricity: firstly because water is free and secondly because no harmful emissions are produced. However, water availability is uncertain, and the use of stored water is sometimes restricted by demands for irrigation.

  The long-term model described is appropriate for thermal generation units but not for hydro generation, since calculation of the latter requires additional variables to represent the variability of water storage in reservoirs and discharges.

  A coarse model of hydro generation is employed. It considers the whole or a part of the reservoir systems of one or several basins as a single pseudo-thermal unit with null cost, outage probability equal to 0 and a large capacity. Constraints are then included binding the hydro generation over successive intervals, so that they add up to a total expected hydro generation for the whole subperiod (usually every three months). This data is deduced from historical records.

  The constraint added for each period of time $\iota \in \mathcal{I}$, where $\mathcal{I}$ is a set of successive intervals in $\{1, \ldots, n_i\}$, is:

$$\sum_{i \in \iota} \sum_{k \in \mathcal{H}} x_k^i \leq \kappa_{\mathcal{H}}^{\iota} \quad \forall \, \iota \in \mathcal{I} \tag{2.7}$$

  where $\mathcal{H}$ is the set of hydro units and $\kappa_{\mathcal{H}}^{\iota}$ the maximum expected hydro generation over a set of intervals ι for which the hydro generation is accounted.

  Pumped storage plants can be modeled more precisely. Following the article of Bloom and Gallant [7], which uses the example of Conejo [14], a storage unit is represented by two pseudo-units: one which represents the charging side and another which is the generator. The potential charging load, which is new demand, is represented by shifting the origin of the LDC in the negative direction by the amount of its charging capacity. Some extra constraints are required to represent the energy balance of the storage plants. In the test cases generated, no pumped storage was considered.

- Bonus-scheme coal

  Some generation units which burn national coal receive a reduction on the coal cost according to a government Act. Each incentivized unit is split into two pseudo-units, with the same capacity but different generation costs. There is a threshold above which exceeding production receives no government subsidy. Two type of constraint are required:

- one that limits the incentivized production within the year,
- another that limits the duration of the generation of the two new units, not to exceed the interval duration.

- Minimum generation time

  Generation units in the Spanish system are paid for their availability. This incentive is designed to guarantee supply during high demand periods, when the most inefficient (and therefore expensive) units may be required. The maintenance of such units would not be profitable if this overhead were not paid, and the load matching would not be guaranteed.

  A unit receives this amount if it has generated a load equivalent to at least 460 hours at maximum capacity during the preceding year. This condition must be accomplished quarterly. This constraint is active for the most inefficient units with low production.

- SGC market-share

  In long-term planning it is reasonable to consider the SGC market share.

- Other constraints

  The SGC may require specifications such as a take-or-pay contract. Other types of constraints are environmental limits, given that CO_2 emissions depend greatly on the fuel used for electricity generation.

All of these constraints are expressed as linear inequality constraints. Parameter n_c is the number of non-LMCs. Matrix $A \in \mathbb{R}^{n_c \times n_u \cdot n_i}$ contains the coefficients for the non-load-matching constraints and vector $a \in \mathbb{R}^{n_c}$ the rhs. Some of the constraints refer to a single interval and others are multi-interval.

2.6 The generation duration curve

The generation duration curve (GDC) is the expected production of 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.

Figure 2.1 shows the GDC for a solution that matches a given LDC (dashed line). The areas under the LDC and under the GDC must coincide, although they do not have the same shape due to the outage probabilities.

The peak power of the GDC is $\sum_{j \in \Omega} c_j + \overline{p}$, and the area above power $\sum_{j \in \Omega} c_j$ is the external energy.

Usually, the GDC is not plotted because the exact position of the splitting has to be calculated. An example of splitting in the solution of figure 2.1 can be found in unit u2, which is split by unit u5.

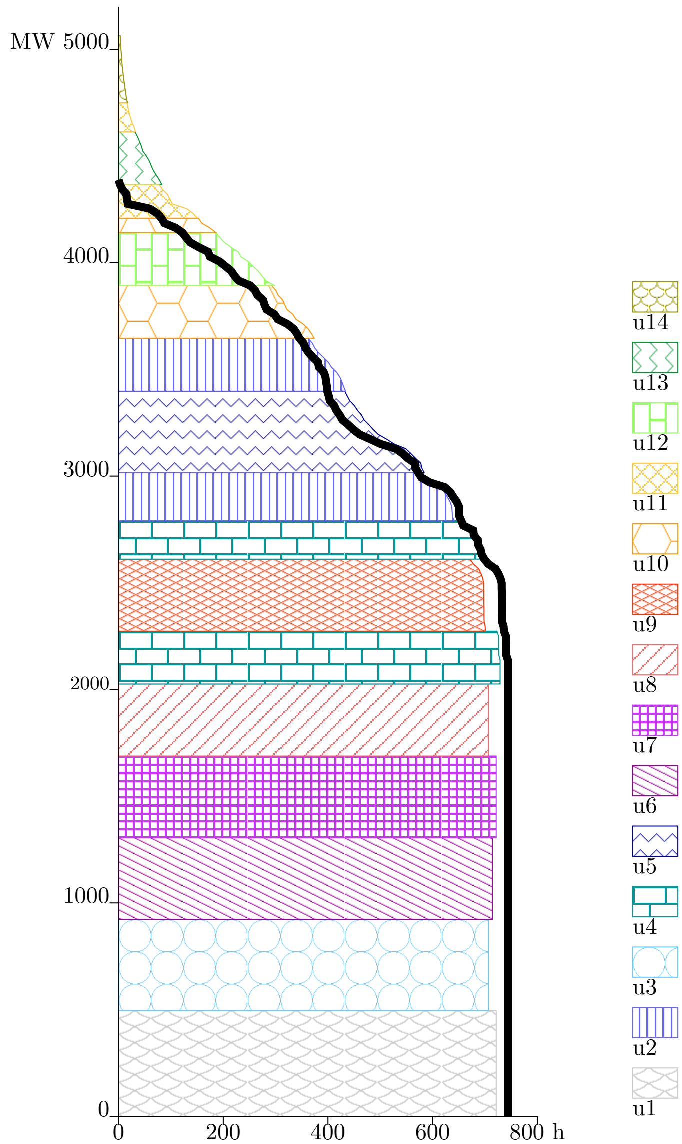


Figure 2.1: Generation-duration curve and load-duration curve (thick line) for a weekly interval in a 14-unit problem.

2.6.1 External energy

The probability of not always serving the entire demand is not null. In such an event, external utilities connected to the pool would supply the part of the production that was lacking. The external energy is represented by the index $n_u + 1$. The *load-balance* equation is always binding:

$$\sum_{j=1}^{n_u+1} x_j = \widehat{e} \tag{2.8}$$

The generation duration curve includes the use of *reserve* units in the pool, as shown in figure 2.1 for units whose layer of generation is above the peak load of the LDC and up to the power $\sum_{j=1}^{n_u} c_j$, which is the total capacity of the pool. Beyond this power, the small and often negligible area of the GDC is the *external energy* that cannot be supplied by the units of the pool. The external energy is either supplied by a neighboring interconnected pool (at the price of non-agreed exchanges) or is not supplied to customers (implying a loss of revenue equivalent to a cost incurred for estimated tariff payment by the pool). We assume that the external energy is always available and its cost is higher than the most expensive unit of the pool.

In the case of a liberalized electricity market that is an electrical island (no connections with other pools or companies), external energy also exists. It is the amount of energy unserved to customers due to accumulations of unit outages that may occur during certain time lapses. The price of this "external energy" is then the tariff paid by customers, as this unserved energy brings about a decrease in revenue for the whole pool. Equation (2.8) is thus the same whether or not the pool is an electrical island.

2.7 The long-term market-price function

In liberalized markets, generation companies must bid their generation to the market operator and a market price is determined every hour by matching the demand with the lowest-priced bids [13]. Generation companies are thus no longer interested in generating power at a lower cost, but in obtaining maximum profit, which is given by the difference between the revenue at market price and the generation cost of any bids accepted. This definition corresponds to the social welfare maximization.

2.7.1 Market-price function with respect to load duration

From the records of past market prices and load series (see figure 2.2) it is possible to compute a market-price function with respect to load duration for each interval. This function is to be used with expected generations that match the LDC of the interval, so the duration of market-price functions correspond with that of the intervals.

Both the load and the corresponding market price are reordered in decreasing load order, giving a price-duration curve for each LDC. The price-duration curve obtained may be non-smooth and non-decreasing. However, it shows a decreasing trend (see figure 2.3) which is adjusted by a linear function: $b^i + l^i t$, t being the load duration. The oscillations (or uncertainty) of market price are smoothed out by considering the linear regression employed, which relates market price to load duration.

The market-price linear function obtained through regression should resemble the *market-price duration curve* described in section (2.3) and proposed in [4].

In order to determine the maximum-profit objective function, a simplifying assumption is suitable regarding the shape of unit contributions in the generation-duration curve (see figure 2.4). Instead of showing some units (particularly those with the lowest loading order) with an irregular shape in its right-hand side, the contribution of all units will be assumed to have a rectangular shape with height c_j (the capacity of unit j) and base length x_j^i/c_j (the expected generation duration) as in figure 2.4.

The profit (revenue at market price minus cost) of unit j in interval i is computed as:

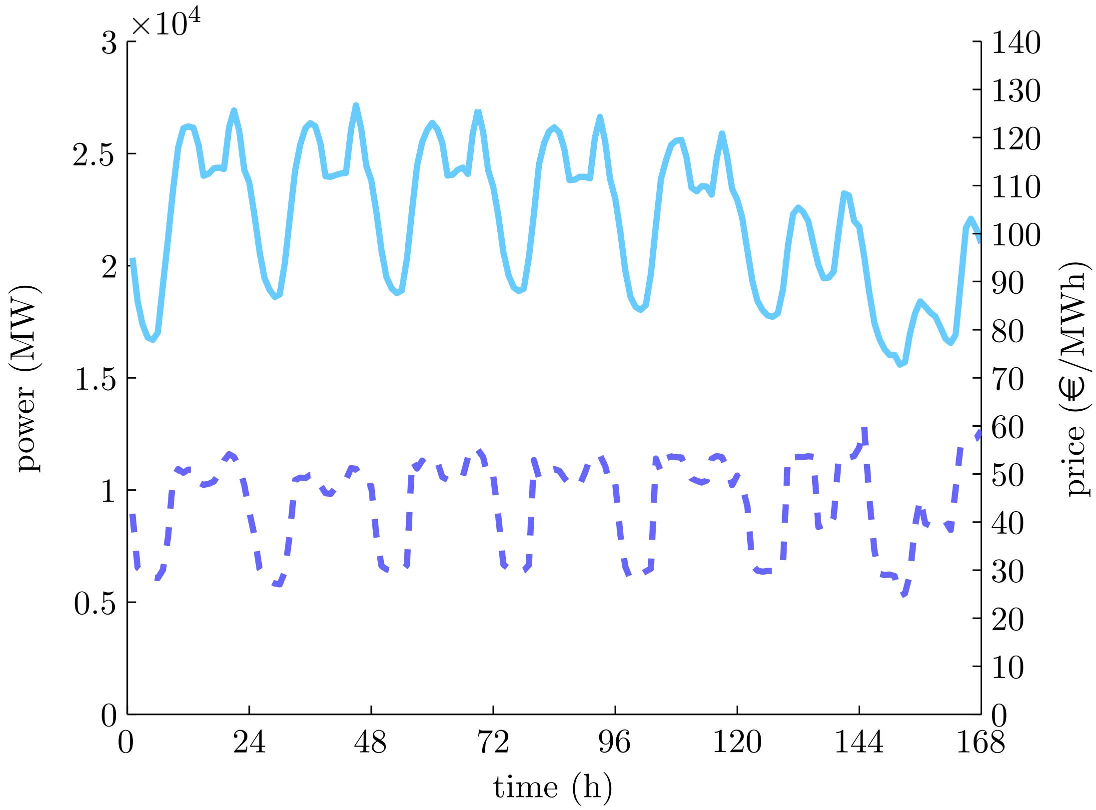


Figure 2.2: Hourly loads (continuous curve) and market prices (dashed)

$$W(x_j^i) = \int_0^{x_j^i/c_j} c_j\{b^i + l^i t - f_j\}dt = (b^i - f_j)x_j^i + \frac{1}{2}\frac{l^i}{c_j}{x_j^i}^2. \tag{2.9}$$

Adding all the intervals and units, and taking into account the external energy, the profit function to be maximized is:

$$\sum_i^{n_i}\left[\sum_j^{n_u}\{(b^i - f_j)x_j^i + \frac{1}{2}\frac{l^i}{c_j}{x_j^i}^2\} - f_{n_u+1}x_{n_u+1}^i\right]$$

which is quadratic in the energies generated. From the load-balance equation (2.8), the external energy can be isolated, and the objective function is simplified:

$$\sum_i^{n_i}\left[\sum_j^{n_u}\{(b^i - f_j + f_{n_u+1})x_j^i + \frac{1}{2}\frac{l^i}{c_j}{x_j^i}^2\} - f_{n_u+1}\widehat{e}^i\right] \tag{2.10}$$

where $-\sum_i^{n_i} f_{n_u+1}\widehat{e}^i$ is a constant.

The profit function to be maximized is quadratic for the expected generation energies. Given that $l^i < 0$, the quadratic part of the objective function (2.10) is negative definite, thus it has a unique

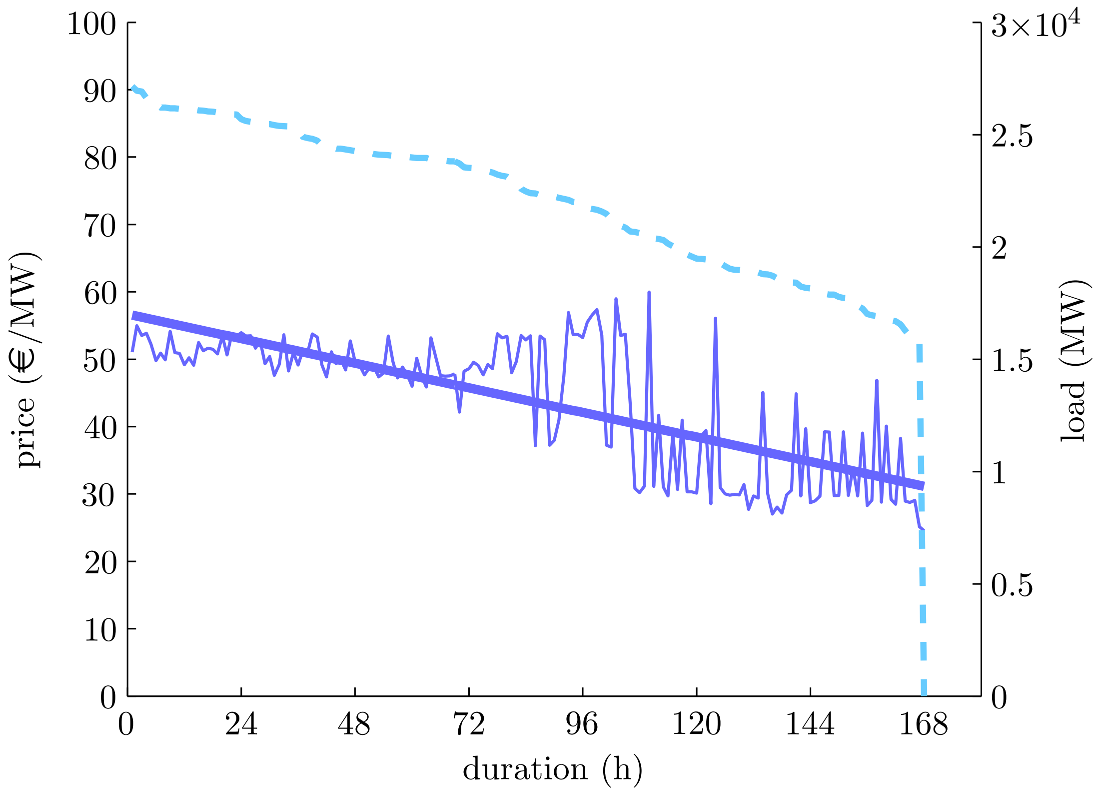


Figure 2.3: Market prices ordered by decreasing load power (thin continuous curve) in weekly intervals, market-price linear function (thick line) and LDC (dashed).

global maximizer.

When the problems are stated in general notation, the minimization form of the objective function is used. The problem is equivalent because maximizing a function is equivalent to minimizing the same function with the sign changed.

For short, we write minus the function (2.10) as $h'x + \frac{1}{2}x'Hx$, where $h \in \mathbb{R}^{n_u \cdot n_i}$ with $h^i_j = f_j - f_{n_u+1} - b^i$ (which is negative) and $H \in \mathbb{R}^{(n_u \cdot n_i) \times (n_u \cdot n_i)}$ is a diagonal matrix with $H^i_j = -l^i / c_j$ (which is positive).

2.7.2 Profit maximization versus other approaches

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 with respect to load duration, and it is considered here that the predicted market-price variation is not altered by generations during the interval; equivalently, it is considered 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

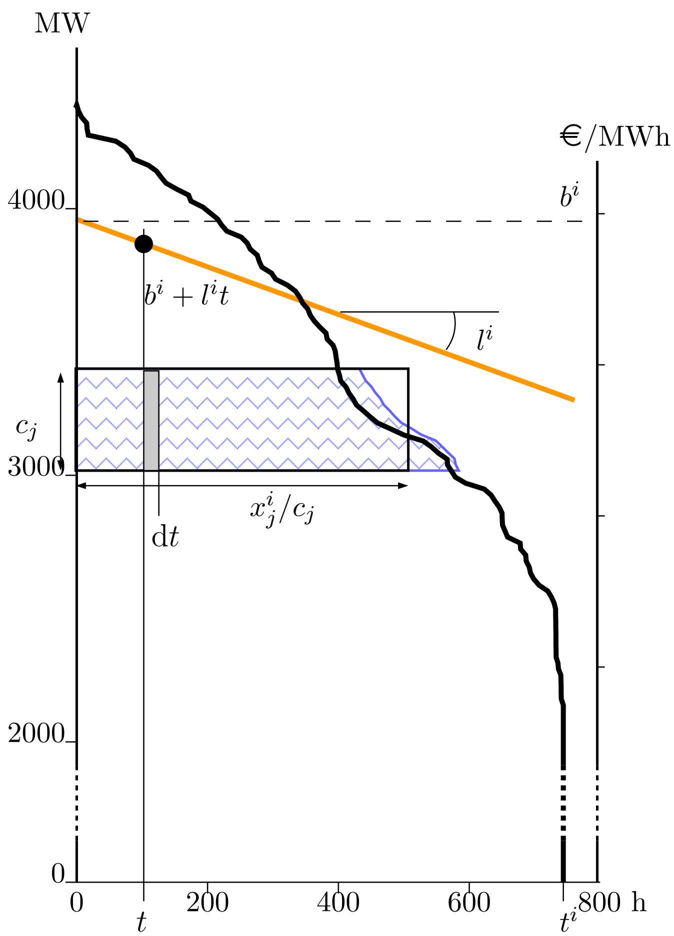


Figure 2.4: Long-term price function for a time interval and contribution of unit j.

correlation can easily be observed. From figure 2.3 it is clear that the linear market-price change averages out the hourly changes of the market price-load ratio.

The market-price function proposed comes from regression with records of historical market-price data, therefore it captures already strategic practices by participants. The influence of the different agents on the market price is stronger in the short term than in the long because, leaving aside collusion, generation with the more efficient units cannot be withheld for a long time lapse without leaving the competitors increase their market share and benefits. As observed in the Spanish pool, hydro generation is a special case, and this is dealt with in the following section.

2.8 Change of the market-price function with hydro generation

Examination of historical records reveals a negative correlation in the long term between hydro generation and market price: the higher the hydraulic generation, the lower the price. Figure 2.5 illustrates this correlation with the weekly moving average of the price and hydro generation.

To take into account this effect in the model presented here, a change in the long-term market-

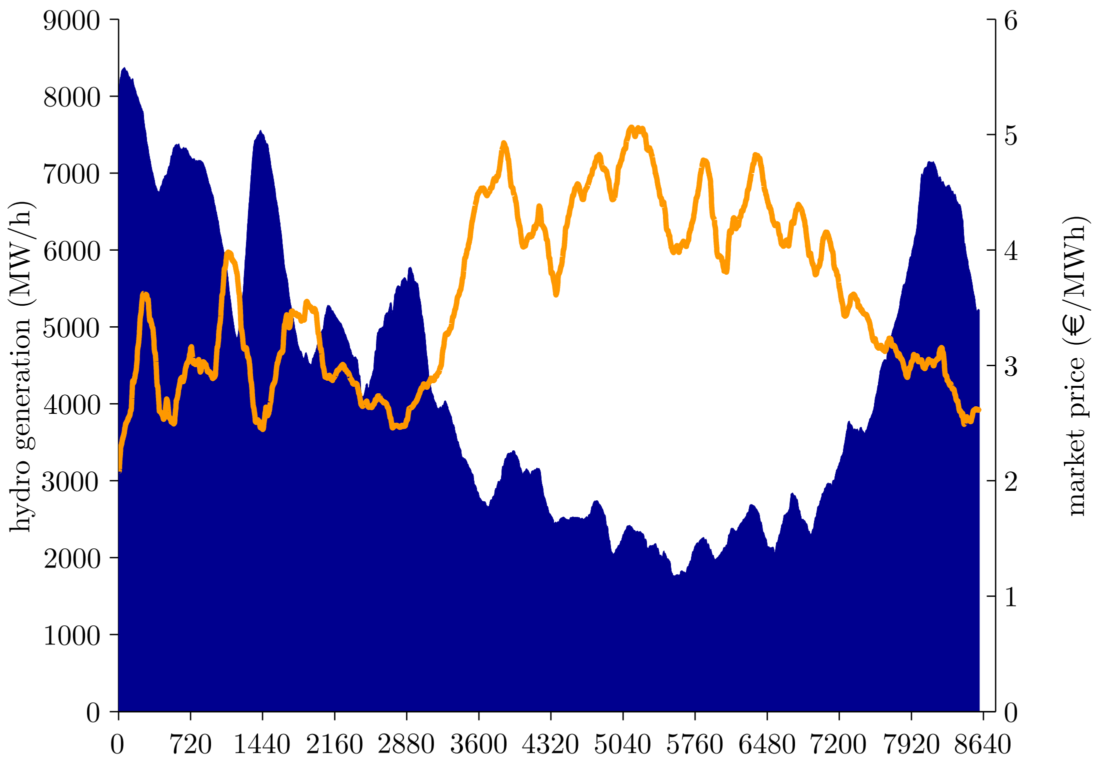


Figure 2.5: Weekly moving average of the market price (orange line) and of the hydro generation (blue area) during the year 2003.

price function is proposed. Given that both the peak and the base power demand prices appear to be equally affected by the hydro generation level, a linear change in the basic coefficient b^i of (2.10) is proposed:

$$b^i = b_0^i - c_0^i \sum_{k \in \mathcal{H}} x_k^i$$

where $\mathcal{H}$ is the set of hydro units and b_0^i and c_0^i are positive coefficients that are estimated from past market price and hydro generation data, taking into account the interval hydro generation and the b^i variation. Integrating as in (2.10) we obtain:

$$\sum_{i}^{n_i} \left[\sum_{j}^{n_u} \{ (b_0^i - f_j + f_{n_u+1}) x_j^i - c_0^i \sum_{k \in \mathcal{H}} x_k^i x_j^i + \frac{1}{2} \frac{l^i}{c_j} {x_j^i}^2 \} - f_{n_u+1} \hat{e}^i \right] . \tag{2.11}$$

This objective function is still quadratic, but its matrix is no longer diagonal and it is indefinite for the values l^i and c_0^i found in practice. The matrix of the objective function is block diagonal, one for each interval i, being the submatrix for interval i:

$$\begin{bmatrix} -l^i/c_1 & & c_0^i & & & & \\ & \ddots & \vdots & & & & \\ c_0^i & \dots & -l^i/c_k + 2c_0^i & \dots & c_0^i & \dots & c_0^i \\ & & \vdots & \ddots & & & \\ & & c_0^i & & -l^i/c_j & & \\ & & \vdots & & & \ddots & \\ & & c_0^i & & & & -l^i/c_{n_u} \end{bmatrix}.$$

where it is assumed that there is only one hydro unit (k) correlated with the price.

2.9 Emission allowance constraints and objective function

The Kyoto Protocol on greenhouse gas emission reduction is in force in the European Union. Compliance with the terms of the protocol has economical effects as member country emissions above or below a given quota must be traded in the EU ETS [61]. The allowances for each country are allotted among those companies whose greenhouse emissions are controlled. The model presented will distinguish between the units of the SGC and those controlled by the rest of the participants (RoP for short).

Volumes of CO_2 emitted are proportional to the generation of each unit. Let γ_j be the coefficient relating energy generated x_j^i to emission, and let $\kappa_{CO_2}^{SGC}$ and $\kappa_{CO_2}^{RoP}$ be the emission allowance allocated to the SCG and RoP units for the optimization period considered. Two new variables are introduced, o_{SGC} and o_{RoP}, which can be positive or negative and represent the unused allowances (when they are positive) or the amount of allowances that will have to be bought (when they are negative) in the EU ETS by the SGC or the RoP, respectively. The equations relating these variables are:

$$\sum_{i}^{n_i} \sum_{j \in \Xi} \gamma_j x_j^i + o_{CO_2}^{SGC} = \kappa_{CO_2}^{SGC} \tag{2.12a}$$

$$\sum_{i}^{n_i} \sum_{j \in \Omega \setminus \Xi} \gamma_j x_j^i + o_{CO_2}^{RoP} = \kappa_{CO_2}^{RoP} \tag{2.12b}$$

where Ξ is the set of SGC units.

Records of emission allowance price [19] appear to be stable for the long term, suggesting that prices can be considered constant for the horizon studied here. The effect on profit of selling unused allowances or of buying the excess at a price ϱ is to add a linear term to the (maximization) objective function:

$$\varrho \cdot (o_{SGC} + o_{RoP})$$

The term $(o_{SGC} + o_{RoP})$ does not necessarily cancel to zero and can be of any sign for a specific power pool.

The emission balance equations (2.12) and variables o_{SGC} and o_{RoP} can be omitted if the variables are isolated from (2.12) and are substituted in the objective function term, contributing in the linear part as:

$$\varrho \cdot \left(\kappa_{CO_2}^{SGC} - \sum_{i}^{n_i} \sum_{j \in \Xi} \gamma_j x_j^i + \kappa_{CO_2}^{RoP} - \sum_{i}^{n_i} \sum_{j \in \Omega \setminus \Xi} \gamma_j x_j^i \right). \tag{2.13}$$

2.10 The long-term power planning problem with maintenance

Thermal units need to be taken out of service for maintenance for a number of weeks each year. Usually this maintenance period is contiguous, although it may also be split into two subperiods. In this thesis the model is limited to the contiguous maintenance-period case, but the formulation presented can be easily extended to the split-period case.

The duration of each unit maintenance is considered known. Let $\Phi \subset \Omega$ be the set of units for which the maintenance planning is optimized, and n_m the cardinality of Φ. Parameter m_j has the number of intervals that unit $j \in \Phi$ needs for maintenance.

A formulation is proposed here for optimizing the selection of the maintenance periods jointly with profit maximization in long-term generation planning.

2.10.1 State of the Art in LTGMP

Conejo *et al.* [10] address a problem akin to the one introduced in this section. Based on their knowledge of the Spanish electricity market, they propose a multi-round mechanism in order to coordinate the maintenance schedules among the SO and the producers so that an appropriate degree of reliability is ensured throughout the year. The time horizon is divided into weeks, and each week consists of six load subperiods. Reliability is measured by comparing the capacity of the non-scheduled units with the predicted load. The iterative procedure works as follows: the SO solves a yearly maintenance schedule for the whole system while maximizing reliability. Each company solves its corresponding maintenance schedule to maximize its profit. Finally, the SO compares its schedule with the ones issued by the companies. While minimum reliability is not ensured, the SO sets up incentives/disincentives for each interval to encourage producers to modify the schedule presented. Our model would be useful for preparing the company maintenance schedule (but does not consider the start-up costs).

Marwali and Shahidehpour [37] propose a complex model involving the units and the transmission-line maintenance schedule. It takes into account crew, system emission limits and fuel constraints. They propose a Benders decomposition to solve their model. The scheduled planning horizon is divided into weeks. The goal is to minimize the total maintenance costs (of units and lines) as well as the cost of purchasing external energy. Although external energy can be purchased the market is not considered. The external energy is computed after matching an Equivalent Load-Duration Curve (ELDC), dispatching the units in merit order for each week. The expected generation of each unit is computed using the Gram-Charlier approximation. Given that the dispatching of fuel and units is decoupled, the ELDC is matched on a merit order basis, and this order is altered iteratively if the fuel dispatch reveals that there are infeasibilities.

A detailed description of a classical maintenance scheduling model can be found in [67], and in the bibliography cited therein.

2.10.2 Model assumptions

The shut-down and start-up costs are not taken into account in our objective function because they are negligible compared to the generation cost value and because they would be a constant, given that maintenance has to be carried out.

When the joint model for generation planning and maintenance schedule is solved, all intervals must have the same length with the exception of the first, which is always one week, and the availability of the units is given.

Maintenance is usually scheduled only for the thermal units of a given SGC, although maintenance of some units controlled by the rest of the participants in the pool can also be optimized. Fixed maintenances for certain other units can also be considered (for example, the nuclear plants).

2.10.3 Variables and maintenance constraints

Three binary variables model the maintenance schedule:

- s_j^i indicates the maintenance $\{0\}$ or in-service $\{1\}$ state of the thermal unit.

- d_j^i indicates whether unit j initiates $\{1\}$ the maintenance or not $\{0\}$ at the beginning of interval i.

- u_j^i indicates whether unit j starts up $\{1\}$ or not $\{0\}$ at the beginning of interval i after maintenance.

The following set of constraints expresses that there will be one maintenance period and only one for each unit $j \in \Phi$ of length m_j intervals.

It should be recalled that the first interval is already fixed and that maintenance must be completed within the time horizon of study.

$$\sum_{i=2}^{n_i} s_j^i = n_i - m_j - 1 \qquad \forall\, j \in \Phi \qquad (2.14)$$

$$\sum_{i=2}^{n_i - m_j} d_j^i = 1 \qquad \forall\, j \in \Phi \qquad (2.15)$$

$$d_j^i + \sum_{l=i}^{i+m_j-1} u_j^l \leq 1 \qquad 2 \leq i \leq n_i - m_j + 1 \qquad \forall\, j \in \Phi \qquad (2.16)$$

$$d_j^i - u_j^{i+m_j} = 0 \qquad 2 \leq i \leq n_i - m_j \qquad \forall\, j \in \Phi \qquad (2.17)$$

$$s_j^i - s_j^{i-1} + d_j^i - u_j^i = 0 \qquad 2 \leq i \leq n_i \qquad \forall\, j \in \Phi \qquad (2.18)$$

Equations (2.14) state that there will be m_j intervals in maintenance for each unit $j \in \Phi$. Equations (2.15) ensure that unit j will stop between the second and the $(n_i - m_j)$th interval, and that there will be only one stop. Therefore, a unit cannot start up while it is undergoing maintenance (2.16). In fact, a unit is always available except when it is stopped for maintenance. A unit starts up just after m_j maintenance intervals, as stated in equations (2.17). Finally, equations (2.18) link the three binary variables.

A unit can generate only when it is in service. This information is given by variable s:

$$x_j^i \leq \overline{x}_j^i s_j^i \qquad \forall\, i \quad \forall\, j \in \Phi \qquad (2.19)$$

This constraint joins the two formulations: generation planning and the maintenance schedule.

There are also certain variables that can be fixed beforehand:

$$d_j^i = 0 \qquad n_i - m_j + 1 \leq i \leq n_i \qquad \forall\, j \in \Phi \qquad (2.20)$$

$$u_j^i = 0 \qquad 1 \leq i \leq m_j + 1 \qquad \forall\, j \in \Phi \qquad (2.21)$$

These constraints arise from the observation that a unit cannot stop during the last intervals because maintenance has to be completed within the time horizon (equations 2.20), and that a unit cannot resume activity before stopping, so variable u cannot be applied to the first intervals.

There are other technical constraints that prevent solutions involving a number of units undergoing maintenance during the same interval.

$$\sum_{j \in \Omega^i \setminus \Phi} c_j + \sum_{j \in \Phi} c_j s_j^i \geq \kappa_s \overline{p}^{\,i} \qquad \forall\, i \qquad (2.22)$$

$$\sum_{j \in \Phi} c_j s_j^i \geq \kappa_c \widehat{p}_c \qquad \forall\, i \qquad (2.23)$$

Equations (2.22) ensure the capacity of the system. The sum of the capacity of the available units is at least κ_s per cent higher than the peak load ($\overline{p}^{\,i}$). Similarly, equations (2.23) request that at least κ_c per cent of the company capacity ($\widehat{p}_c$) be available at all times. Usually κ_s is 20 %, and κ_c varies from 50% to 75%.

2.11 Mathematical models

This section presents the models that will be used in the following chapters.

2.11.1 The long-term generation planning problem

The following model is the long-term generation planning problem, LTGP problem for short:

$$\underset{x}{\text{maximize}} \quad \sum_{i}^{n_i} \sum_{j}^{n_u} \{(b^i - \widetilde{f}_j)x_j^i + \frac{1}{2}\frac{l^i}{c_j}\, {x_j^i}^{2}\} \tag{2.24a}$$

$$\text{subject to} \quad \sum_{j\in\omega} x_j^i \le \widehat{e}^{\,i} - w^i(\omega) \qquad \forall\, i \qquad \forall\, \omega \subseteq \Omega \tag{2.24b}$$

$$Ax \ge a \tag{2.24c}$$

$$x_j^i \ge 0 \qquad \forall\, i \qquad \forall\, j \in \Omega \tag{2.24d}$$

Constant $-\sum_i^{n_i} f_{n_u+1}\widehat{e}^i$ is to be added to the objective function (2.24a) in order to compute the actual value (2.10). Parameter $\widetilde{f}_j$ stands for the generation cost minus the price of the external energy ($f_j - f_{n_u+1}$).

Constraints (2.24b) constitute the extension to the multi-interval case for LMCs (2.5). Constraints (2.24c) are the non-LMCs explained in section 2.5.

LTGP with hydro generation influence

In the event that the influence of hydro generation on market prices were considered, the following model would be solved:

$$\max_{x} \sum_{i}^{n_i} \left[\sum_{j}^{n_u} \{(b_0^i - \widetilde{f}_j)x_j^i - c_0^i \textstyle\sum_{k\in\mathcal{H}} x_k^i x_j^i + \frac{1}{2}\frac{l^i}{c_j}\, {x_j^i}^{2}\} \right] \tag{2.25a}$$

$$\text{s.t.} \quad \sum_{j\in\omega} x_j^i \le \widehat{e}^{\,i} - w^i(\omega) \qquad \forall\, i \qquad \forall\, \omega \subseteq \Omega \tag{2.25b}$$

$$Ax \ge a \tag{2.25c}$$

$$x_j^i \ge 0 \qquad \forall\, i \qquad \forall\, j \in \Omega \tag{2.25d}$$

This model will be referred to as $\text{LTGP}_{\mathcal{H}}$.

Constant $-\sum_i^{n_i} f_{n_u+1}\widehat{e}^i$ is to be added to the objective function (2.25a) in order to compute the actual value (2.11). Parameter $\widetilde{f}_j$ stands for the generation cost minus the price of the external energy $(f_j - f_{n_u+1})$. The constraints are the same as for the LTGP problem.

2.11.2 Model of joint long-term generation planning and unit maintenance

The following model is the formulation proposed for planning long-term generation together with the schedule of maintenance periods for the thermal units. This model will be referred to herein as the LTGMP problem.

$$\max_{x,s,d,u} \sum_i^{n_i}\sum_j^{n_u}\{(b^i - \widetilde{g}_j)x_j^i + \tfrac{1}{2}\tfrac{l^i}{c_j}\, {x_j^i}^2\} \quad (2.26a)$$

$$\text{s. t.} \sum_{j\in\omega} x_j^i \le \widehat{e}^i - w^i(\omega) \qquad \forall\, i \qquad \forall\, \omega^i \subseteq \Omega \quad (2.26b)$$

$$Ax \ge a \quad (2.26c)$$

$$\sum_{i=2}^{n_i} s_j^i = n_i - m_j - 1 \qquad \forall\, j \in \Phi \quad (2.26d)$$

$$\sum_{i=2}^{n_i-m_j} d_j^i = 1 \qquad \forall\, j \in \Phi \quad (2.26e)$$

$$d_j^i + \sum_{l=i}^{i+m_j-1} u_j^l \le 1 \qquad 2 \le i \le n_i - m_j + 1 \qquad \forall\, j \in \Phi \quad (2.26f)$$

$$d_j^i - u_j^{i+m_j} = 0 \qquad 2 \le i \le n_i - m_j \qquad \forall\, j \in \Phi \quad (2.26g)$$

$$s_j^i - s_j^{i-1} + d_j^i - u_j^i = 0 \qquad 2 \le i \le n_i \qquad \forall\, j \in \Phi \quad (2.26h)$$

$$d_j^i = 0 \qquad n_i - m_j + 1 \le i \le n_i \qquad \forall\, j \in \Phi \quad (2.26i)$$

$$u_j^i = 0 \qquad 1 \le i \le m_j + 1 \qquad \forall\, j \in \Phi \quad (2.26j)$$

$$\sum_{j\in\Omega^i\setminus\Phi} c_j + \sum_{j\in\Phi} c_j s_j^i \ge \kappa_s \overline{p}^{\,i} \qquad \forall\, i \quad (2.26k)$$

$$\sum_{j\in\Phi} c_j s_j^i \ge \kappa_c \widehat{p}_c \qquad \forall\, i \quad (2.26l)$$

$$x_j^i \le \overline{x}_j^i s_j^i \qquad \forall\, i \qquad \forall\, j \in \Phi \quad (2.26m)$$

$$x_j^i \le \overline{x}_j^i \qquad \forall\, i \qquad \forall\, j \in \Omega \setminus \Phi \quad (2.26n)$$

$$s_j^i, d_j^i, u_j^i \in \{0,1\} \qquad \forall\, i \qquad \forall\, j \in \Phi \quad (2.26o)$$

$$x_j^i \ge 0 \qquad \forall\, i \qquad \forall\, j \in \Omega \quad (2.26p)$$

The same comments from the previous section apply to equations (2.26a – 2.26c). Equations (2.26c – 2.26l) model the maintenance schedule as developed in section 2.10. Constraint (2.26m) limits the maximum generation according to the state of the unit for those units whose maintenance is to be placed optimally; for the rest, a simple bound is added (2.26n). The maintenance schedule is modeled with binary variables (s, d and u) and generation planning is defined using continuous variables (x).

It should be noted that the model (2.26) put forward includes all units in the pool and matches the pool LDCs of the successive intervals. Normally the only units whose maintenance schedule is optimized are the units of a specific SGC. The maintenances of units of other participants in the pool are either fixed, if known, or taken into account approximately by restricting their production over some intervals (where maintenances usually take place).

2.12 Scope of this thesis for modeling stochasticity and market equilibrium

Several choices had to be made regarding the overall scope of this thesis and the amount of work required to complete it.

The main idea was to formulate basic models for LTGP in liberalized markets (of the type existing in Spain in 2003) leading to optimization problems and to develop efficient solution techniques.

With reference to stochasticity:

- For planning the generation, only the expected generation for the whole interval was used in this thesis. The use of LDCs, made up of hourly loads, means that the variability of the demand within the interval is represented accurately.

- The use of a predicted LDC for each interval, the Bloom and Gallant formulation, the unit outage probabilities, plus the convolution method for updating the survival function of the load, partly accounts for load variability and fully accounts for unit outages and for the load matched by expected generation.

- The use of an exogenous market price function for each interval, obtained through a regression of price with respect to load, partly accounts for market price uncertainty. The extension to an endogenous formulation that takes into consideration the effect of hydro generation means that the model employed is closer to the real patterns of markets in which hydro has an important share.

- Employing market share constraints for an SGC partly accounts for the risk of losing revenue.

- Extending the models in order to take into account the stochasticity would require the use of scenarios and are considered to be beyond the scope of this thesis. Possible examples are:

  - the stochasticity in hydro generation, which would require consideration of several scenarios in the κ_H of the rhs in (2.7).
  - the representation of hydro generation through an interval-replicated hydronetwork with detailed reservoirs, natural inputs in each interval and water balance equations as described in [45].

- the stochasticity in load (or in the change of the LDC due to wind-power generation) considering different scenarios, each with an LDC of different total energy $\widehat{e}^i$.
- the stochasticity in market price, which would require consideration of several scenarios for the basic price b^i for the exogenous model and b_0^i for the endogenous model taking into account the effect of hydro generation.
- use of value-at-risk constraints to ensure prevent an excessive loss of profit.

With reference to market equilibrium:

- The profit maximization (welfare) employed is an imperfect equilibrium, because each market participant could still change its generation so as to increase its profit (provided that the future bids of other participants were known).
- A reasonable market equilibrium to seek is the Nash equilibrium [15], in which the participants would not wish to alter their generation as this would also mean decreasing their own profit. Finding the Nash equilibrium with the market model proposed is beyond the scope of this thesis. It could be calculated through the Nikaido-Isoda relaxation algorithm [62], which consists of successive optimizations with changing objective functions, where the first of which corresponds to the profit maximization employed.

Finally, in reference to liberalized market types, the Spanish electricity market recently (July 2006) introduced bilateral contracts and futures trading, but until that point it was an entirely bid-based market. This thesis is based on the latter type, as the data available for checking the models proposed was taken from the Spanish pool.

Chapter 3

Column-generation methods

The long-term generation planning problem using the Bloom and Gallant formulation can be solved with several methods. A direct approach (QP or IPQP) is of no use due to the exponential number of LMCs. The inconvenience does not come only from the storage requirement but also from the complexity of computing the corresponding right-hand side terms of (2.5), which is very time consuming. These computations require numerical integration and convolutions of the load survival function. Moreover, only a few of them (at most $n_i \times n_u$ out of $n_i \times (2^{n_u} - 1)$) will be active at the optimizer.

This chapter is devoted to the solution of the LTGP problem using a decomposition method applied to a transformation of the problem, which is an optimal procedure and does not require the computation of all the LMC right-hand sides.

3.1 Decomposition methods: Dantzig-Wolfe

Decomposition techniques solve large problems with a particular structure, which is exploited both theoretically and computationally, through iterative solutions of smaller problems with a similar structure. Decomposition is desirable when the subproblems are much easier to solve than the full problem.

The problem being solved is split into two problems: the master problem and the subproblem. The master problem is the original problem with only a subset of variables being considered. The subproblem is a new problem created to identify a new variable.

As early as 1958 Ford and Fulkerson [20] suggested using implicit variables of a multi-commodity network flow problem. By 1960, Dantzig and Wolfe [17] had adapted it to linear programming problems with a decomposable structure and established it as a methodology.

Dantzig-Wolfe decomposition, also called column generation, is applied when the problem concerned has a large number of variables and only a small subset will be different from 0 at the optimizer, therefore a small subset needs to be considered when solving the problem. Column generation adds to the master problem the variables that could potentially improve the objective function.

The process works as follows. First, the master problem is solved. From this solution, the dual prices for each of the constraints in the master problem are computed. This information is sent to the objective function of the subproblem. The subproblem is then solved. If the objective value of the subproblem is negative, a variable with negative reduced cost has been identified. This variable is then added to the master problem and the master problem is re-solved. Re-solving the master problem will generate a new set of dual values and the process is then repeated until no negative reduced cost variables are identified. When the subproblem returns a solution with non-negative reduced cost, it can be concluded that the solution to the master problem is optimal.

3.2 Transformation of the LTGP problem

The transformation of the LTGP problem is based on the observation that the load-matching constraints (2.5) and the non-negativity bounds on the expected generations define for each interval a convex polyhedron whose vertices can be easily calculated.

3.2.1 Defining a vertex of the convex polyhedron as the merit order solution for a given set of linear generation costs

Let us consider the linear problem:

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \widetilde{f}^{i\,\prime} x^i \\ \text{subject to} & \sum\limits_{j \in \omega} x^i_j \leq \widehat{e}^i - w^i(\omega) \quad \forall\, \omega \subseteq \Omega \\ & x^i_j \geq 0 \quad \forall\, j \end{array} \tag{3.1}$$

for a given cost vector $\widetilde{f}^i$.

The solution of (3.1) will be a vector V^i of the polyhedron, and every vertex of the polyhedron corresponds to the solution of (3.1) for a certain $\widetilde{f}^i$.

In order to obtain the solution to (3.1) it is not necessary to solve the linear program (3.1). It can easily be shown that the merit order loading of the units according to the costs in $\widetilde{f}^i$ produces the optimizer of (3.1) [43]. This reduces the problem to solving the merit order loading system that, assuming $\widetilde{f}^i_j \leq \widetilde{f}^i_{j+1} \forall\, j$, is:

$$\begin{bmatrix} 1 & & & & \\ 1 & 1 & & & \\ 1 & 1 & 1 & & \\ 1 & 1 & 1 & \ddots & \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} x^i_1 \\ x^i_2 \\ x^i_3 \\ \vdots \\ x^i_{n_u} \end{bmatrix} = \begin{bmatrix} \widehat{e}^i - w^i(\{1\}) \\ \widehat{e}^i - w^i(\{1,2\}) \\ \widehat{e}^i - w^i(\{1,2,3\}) \\ \vdots \\ \widehat{e}^i - w^i(\Omega) \end{bmatrix}$$

which, in turn, is equivalent to calculating the successive convolutions as in (2.1)

$$x_{j|\Omega_{o_j}} = t(1 - q_j) \int_0^{c_j} S_{\Omega_{o_j}}(z)\, dz \quad \forall\, j$$

where Ω_{o_j} is the subset of unit indices ordered before unit j according to the ordering Ω_o, which is the merit order for $\widetilde{f}^i$.

3.2.2 Feasible points

Any feasible point, in terms of LMCs, can be expressed as a convex combination of all the vertices of each interval i:

$$x^i = V^i \lambda^i \tag{3.2a}$$

$$e' \lambda^i = 1 \tag{3.2b}$$

$$\lambda^i \geq 0 \tag{3.2c}$$

where $V^i \in \mathbb{R}^{n_u \times n_V^i}$ is the set of vertices defined by the LMCs of interval i and e is the all-one vector of appropriate dimension.

The number of vertices of each polyhedron is also high, as is the number of constraints (2.5) that define it. Note that no provision is made for extreme rays, as the nature of the constraints and nonnegativity bounds prevents their occurrence.

3.2.3 Problem transformation

The LTGP problem (2.24) is of type

$$\begin{array}{rl} \text{minimize} & h'x + \frac{1}{2}x'Hx \\ \text{subject to} & Ax \geq a \\ & Bx \leq r \\ & x \geq 0 \end{array} \tag{3.3}$$

where matrix B has the load-matching constraints. Subtracting surpluses $\sigma \in \mathbb{R}^{n_c}$ for the non-LMC inequalities, the transformation of problem (3.3) is rewritten as:

$$\begin{array}{rll} \underset{\lambda,\sigma}{\text{minimize}} & h'V\lambda + \frac{1}{2}\lambda' V' H V \lambda & \\ \text{subject to} & AV\lambda - I\sigma = a & \\ & e'\lambda^i = 1 & \forall\, i \\ & \lambda \geq 0, \sigma \geq 0 & \end{array} \tag{3.4}$$

which is quadratic in λ and lends itself to solution with a column-generation procedure using the Murtagh and Saunders (M&S) algorithm [40, 25]. Matrix I is the identity matrix of appropriate dimension. Let $\psi(\lambda) = h'V\lambda + \frac{1}{2}\lambda' V' H V \lambda$ denote the objective function.

3.3 Murtagh and Saunders algorithm using column generation

Given that the problem (3.4) has a quadratic objective function with linear constraints, the Murtagh and Saunders algorithm [40] is applied together with the column-generation procedure. The outline of the method is as follows:

Murtagh and Saunders algorithm

i Given an initial feasible point λ, σ, classify each variable as basic, superbasic or nonbasic.

Compute the projected gradient, $\|Z'\nabla\psi\|$

ii If $\|Z'\nabla\psi\| \leq \epsilon_g$

- Compute the Lagrange multipliers ν of the active nonnegativity bounds.
- If there is any constraint with a negative multiplier $\nu_l < 0$ then
    - Update $\|Z'\nabla\psi\|$

  otherwise
    - END

iii If $\|Z'\nabla\psi\| > \epsilon_g$

- Compute a descent direction for the basic and superbasic variables, Δ.
- Determine the step length, α.
- Update the basic and nonbasic variables:
    - $\lambda := \lambda + \alpha\Delta_\lambda$
    - $\sigma := \sigma + \alpha\Delta_\sigma$
- Update the basic, superbasic and nonbasic sets.
- Update $\|Z'\nabla\psi\|$.
- Go to step (ii).

3.3.1 Variable classification

In an active set methodology, such as that of Murtagh and Saunders, the active constraints at a feasible point λ, σ are either general linear constraints or simple bounds.

At a typical iteration of an active set method applied to (3.4), the matrix of active constraints $\widehat{M}$ will contain all the general linear constraints, M, and an additional set of rows for the identity matrix that corresponds to variables at zero.

$$\widehat{M} = \left[\begin{array}{c|c} \multicolumn{2}{c}{M} \\ \hline & I \end{array}\right]$$

The Murtagh and Saunders algorithm classifies variables as either *basic*, *superbasic* or *nonbasic* according to the relative position of their value and bounds. Superbasic variables have values strictly between their bounds, whereas the nonbasic variables have their value fixed on a bound (to 0, in our case). The basic variables are chosen such that the basic matrix is square and non-singular; some of these may be on one of its bounds. Next, the constraint matrix (transformed non-LMCs and convexity constraints), M, is (conceptually) partitioned as follows:

$$M_B = \left[\begin{array}{c|c} AV_B & -I_B \\ \hline e' \quad e' & \mathbf{0} \end{array}\right] \qquad M_S = \left[\begin{array}{c|c} AV_S & -I_S \\ \hline e' \quad e' & \mathbf{0} \end{array}\right] \qquad M_N = \left[\begin{array}{c|c} AV_N & -I_N \\ \hline e' \quad e' & \mathbf{0} \end{array}\right]$$

Matrix I stands for the identity matrix. When this has a subindex $_B$, $_S$ or $_N$ corresponds to those columns of the identity matrix belonging to the surpluses σ that have been classified in such group.

Matrix $M_B \in \mathbb{R}^{n_B \times n_B}$ is the basic matrix, whose columns correspond to the basic variables. The n_N columns of M_N correspond to the nonbasic variables (those fixed at 0). The $n_S = (n_V + n_c) - n_B - n_N$ columns of the M_S matrix correspond to the remaining variables.

The number of superbasic variables is a measure of the nonlinearity of the problem: the fewer superbasic variables, the more linear the problem. The limit is when the problem is linear and the M&S algorithm works as the simplex method, in which case there are only basic and nonbasic variables.

As we are dealing with quadratic programming, there is no a priori number of fixed (nonbasic) variables to 0, but the number of superbasic variables has to be smaller than the number of non-linear variables.

3.3.2 The projected gradient

A necessary, but not sufficient, condition for optimality is for the projected gradient to vanish:

$$\|Z'\nabla\psi\| \leq \epsilon_g.$$

The matrix of the null space Z, such that $\widehat{M}Z = 0$, is defined as

$$Z = \begin{bmatrix} -M_B^{-1}M_S \\ I \\ \mathbb{0} \end{bmatrix}. \tag{3.5}$$

Matrix Z has $n_V + n_c$ rows and n_S columns. Neither the inverse of the basis M_B nor the matrix Z are computed explicitly.

Vector $\nabla\psi$ is the gradient of the objective function. For the quadratic function of the transformed problem, the gradient for each group of variables at a particular point λ, σ is:

$$\begin{aligned} \nabla\psi_{\lambda_B} &= h'V_B + V_B'HV_B\lambda_B + V_B'HV_S\lambda_S + V_B'HV_N\lambda_N & \nabla\psi_{\sigma_B} &= 0 \\ \nabla\psi_{\lambda_S} &= h'V_S + V_S'HV_B\lambda_B + V_S'HV_S\lambda_S + V_S'HV_N\lambda_N & \nabla\psi_{\sigma_S} &= 0 \\ \nabla\psi_{\lambda_N} &= h'V_N + V_N'HV_B\lambda_B + V_N'HV_S\lambda_S + V_N'HV_N\lambda_N & \nabla\psi_{\sigma_N} &= 0 \end{aligned} \tag{3.6}$$

The terms in which λ_N appears vanish because $\lambda_N = 0$. The final expression of the projected gradient is as follows:

$$Z'\nabla\psi = \nabla\psi_S - M'_S M_B^{-1}{}'\nabla\psi_B = \nabla\psi_S - M'_S\pi \tag{3.7}$$

where $\nabla\psi_B$ and $\nabla\psi_S$ refer to the gradient with respect to the basic and superbasic variables, and π comes from solving system $M'_B\pi = \nabla\psi_B$, the Lagrange multipliers of the equalities.

3.3.3 Computation of the multipliers and modified costs

The overdetermined system $\widehat{M}'\begin{bmatrix}\pi\\ \nu\end{bmatrix} = \nabla\psi$ is compatible when $Z'\nabla\psi = 0$. The detailed subsystem $\begin{bmatrix} M'_B & \\ M'_N & I\end{bmatrix}\begin{bmatrix}\pi\\ \nu\end{bmatrix} = \begin{bmatrix}\nabla\psi_B\\ \nabla\psi_N\end{bmatrix}$ is:

$$\left[\begin{array}{c|c|c} (AV_B)' & e \; \cdots \; e & \mathbb{0} \\ \hline -I'_B & \mathbf{0} & \\ \hline (AV_N)' & e \; \cdots \; e & I \\ \hline -I'_N & \mathbf{0} & \end{array}\right] \begin{bmatrix}\pi_A\\ \cdots\\ \pi_\lambda \\ \hline \nu\end{bmatrix} = \begin{bmatrix}\nabla\psi_{\lambda_B}\\ \cdots\\ \nabla\psi_{\sigma_B}\\ \hline \nabla\psi_{\lambda_N}\\ \cdots\\ \nabla\psi_{\sigma_N}\end{bmatrix}$$

The solution procedure first finds the constraint multipliers:

$$M'_B\begin{bmatrix}\pi_A\\ \pi_\lambda\end{bmatrix} = \nabla\psi_B \tag{3.8}$$

where $\pi_A \in \mathbb{R}^{n_c}$ are the multipliers of the transformed non-LMCs and $\pi_\lambda \in \mathbb{R}^{n_i}$ are the multipliers of the convexity constraints. This calculation is already performed in the projected gradient computation (3.7).

Formally, V_k has dimension $n_i \cdot n_u$. However, the part that is different from 0 refers to a specific interval. From the equations that yield the multipliers of the active bounds ν, two possible types of equations follow:

$$\pi_\lambda^i + V_{N_k^i}{}'(A'\pi_A) + \nu_k^i = \nabla\psi_{\lambda_{N_k^i}} \tag{3.9}$$

$$-\pi_{A\,k} + \nu_k = \nabla\psi_{\sigma_{Nk}} \tag{3.10}$$

From (3.9) and using the gradient expression (3.6), the multiplier is:

$$\nu_k^i = [h + HV_B\lambda_B + HV_S\lambda_S - (A'\pi_A)]'V_{N_k^i} - \pi_\lambda^i. \tag{3.11}$$

Therefore, a nonbasic vertex k of interval i, $V_{N_k^i}$, is a candidate to enter the basis if ν_k^i is negative. From the other type of multiplier (3.10) we obtain:

$$\nu_k = \pi_{A\,k} \tag{3.12}$$

which indicates that the surplus $\sigma_{A\,k}$ will become superbasic (relaxing the active constraint A_k) whenever $\pi_{Ak} < 0$.

Obtention of new vertices

The problem of finding a vertex V_k such that

$$\widetilde{f}\,'V_k < \pi_\lambda,$$

where $\widetilde{f}$ are the modified costs $(h + HV_B\lambda_B + HV_S\lambda_S - (A'\pi_A))$ can be performed as follows: the first step is to solve

$$\begin{array}{rl} \text{minimize} & \widetilde{f}^i\,'V_k^i \\ \text{subject to} & B^iV_k^i \leq r^i \\ & V_k^i \geq 0 \end{array}$$

and once we have V_k^i it is necessary to check whether $\widetilde{f}^i\,'V_k^i$ is lower than π_λ^i. Only when $\widetilde{f}^i\,'V_k^i < \pi_\lambda^i$ V_k^i will be taken as a new vertex, otherwise, the process is attempted for a different interval.

As specified in section 3.2.1, the solution of the former linear program is straightforward: sort the elements of $\widetilde{f}^i$ in increasing value determining the loading order, then compute the elements of V_k^i by successive convolution (2.2) and integration (2.1).

The nonavailability of units – by programmed overhauling during the interval – is taken into account in the calculation of vertices.

3.3.4 Finding a descent direction

If the optimizer has not been reached ($\|Z'\nabla\psi\| > \epsilon_g$), another feasible point that decreases the objective function value must be found. As this is a constrained problem, a feasible direction is $\Delta = Zp_Z$, for any p_Z:

$$\Delta = \begin{bmatrix} -M_B^{-1}M_S \\ I \\ \mathbb{0} \end{bmatrix} p_Z = \begin{bmatrix} -M_B^{-1}M_S p_Z \\ I p_Z \\ \mathbb{0} \end{bmatrix} = \begin{bmatrix} \Delta_B \\ \Delta_S \\ \Delta_N \end{bmatrix}$$

where the nonbasic variables do not change their value (in this case, they are fixed at 0). The projected gradient direction, $p_Z = -Z'\nabla\psi'$, can be employed or alternatively the Newton direction:

$$Z'(V'HV)Zp_Z = -Z'\nabla\psi' \tag{3.13}$$

where $V'HV$ is the Hessian matrix of the transformed objective function.

Computational experience from the test cases solved shows that the projected gradient direction has poor convergence. The Newton direction is computationally harder to obtain but is much more efficient. When the step length applied is 1, only one iteration is required to achieve $\|Z'\nabla\psi\| \leq \epsilon_g$. However, in our computational implementation, sometimes more than one iteration was necessary when applying a step length of 1 using the Newton direction.

Solution of linear systems using matrix M_B

In order to find the solution of systems (3.8) and $M_B\Delta_B = -M_S p_Z$ the routine LA05 of the *Harwell* library was employed. This routine permits the updating of the factorization of the basic matrix M_B when a column changes.

The same routine was also applied to solve system (3.13) although it is not too appropriate, given that $Z'(V'HV)Z$ is dense and would require pivoting to obtain accurate solutions. This may be the reason why more than one Newton iteration is necessary to reduce $\|Z'\nabla\psi\| \leq \epsilon_g$.

3.3.5 Computation of the step length

Given a feasible point, λ, σ, and a direction Δ, the new point moves to

$$\lambda := \lambda + \alpha\Delta_\lambda \tag{3.14}$$

$$\sigma := \sigma + \alpha\Delta_\sigma \tag{3.15}$$

where Δ_λ and Δ_σ are the components of Δ related to λ and σ respectively. The optimal step length

$$\alpha^* = \frac{-\nabla\psi_\lambda\Delta_\lambda}{\Delta_\lambda V'HV\Delta_\lambda} \tag{3.16}$$

should be α^* equal to 1 if the Newton direction is used. However, α^* may lay beyond the upper limits due to the basic and superbasic variable change.

The variables must be nonnegative, thus:

$$\overline{\alpha}_B = \min \left\{ \frac{\lambda_B^j}{|\Delta_{\lambda_B}^j|} \forall j \,\middle|\, \Delta_{\lambda_B}^j < 0, \frac{\sigma_B^j}{|\Delta_{\sigma_B}^j|} \forall j \,\middle|\, \Delta_{\sigma_B}^j < 0 \right\} \tag{3.17}$$

$$\overline{\alpha}_S = \min \left\{ \frac{\lambda_S^j}{|\Delta_{\lambda_S}^j|} \forall j \,\middle|\, \Delta_{\lambda_S}^j < 0, \frac{\sigma_S^j}{|\Delta_{\sigma_S}^j|} \forall j \,\middle|\, \Delta_{\sigma_S}^j < 0 \right\} \tag{3.18}$$

The step length is $\alpha = \min\{\overline{\alpha}_B, \overline{\alpha}_S, \alpha^*\}$. Depending on which variable gives α, changes in the basic, superbasic and nonbasic sets may occur.

3.3.6 Changes in the variable sets

Should α be equal to

- α^*, no changes occur in the working set.
- $\overline{\alpha}_B$, a basic variable becomes nonbasic and a variable from the superbasic set changes to basic in its place. In this case, the basis factorization has to be updated.
- $\overline{\alpha}_S$, a superbasic variable becomes zero and changes to the nonbasic set.

In theory, the superbasic chosen to be basic only has to be linearly independent from the remaining basics. In practice, the variable has to be chosen more carefully in order to avoid getting stuck without apparent reason.

3.3.7 Choosing a superbasic variable to enter the basis

The choice of the superbasic to enter the basis is important for solution accuracy and convergence. It is desirable to keep the condition number of M_B as low as possible in order to obtain accurate calculations of π and Δ_B.

Once the basic variable l that leaves the basis is known, the question is which superbasic will perform better. The new basis $\widetilde{M}_B$ will be equivalent to the former one M_B except for the leaving column l. The change using an η matrix can be expressed as

$$\widetilde{M}_B = M_B \eta$$

An η matrix is similar to the unit matrix except for one column. For a change of the basis in column l the η matrix is:

$$\eta = \begin{pmatrix} 1 & & & y_1 & & \\ & 1 & & y_2 & & \\ & & \ddots & \vdots & & \\ & & & y_l & & \\ & & & \vdots & \ddots & \\ & & & y_{n_B} & & 1 \end{pmatrix}$$

where y is the vector that solves $M_B y = M_{Sk}$, M_{Sk} being the entering column of the superbasic set. The calculation of the column k of $M_B^{-1} M_S$ is at no cost if using the Newton direction since the explicit calculation of the upper part of Z is required.

It is easy to find an upper bound for the new basis condition number:

$$cond(\widetilde{M}) \leq cond(M) \cdot cond(\eta)$$

The condition number of a matrix is the ratio of its bigger to its smaller singular value [27]. In the case of an unsymmetric square matrix, as our η matrix is, it is more convenient using the ratio of the square root of the biggest and the smallest eigenvalues of the product of the transpose and the matrix [26]. In our case, $\eta'\eta$ is:

$$\eta'\eta = \begin{pmatrix} 1 & & & y_1 & & \\ & 1 & & y_2 & & \\ & & \ddots & \vdots & & \\ y_1 & y_2 & \cdots & \|y\|^2 & \cdots & y_{n_B} \\ & & & \vdots & \ddots & \\ & & & y_{n_B} & & 1 \end{pmatrix}$$

where $\|y\|$ is the 2-norm of vector y. The computation of its eigenvalues by equating to zero $\det(\eta'\eta - \lambda I)$ yields:

$$\det(\eta'\eta - \lambda I) = (1-\lambda)^{(n_B-2)} \left[\lambda^2 - (1 + \|y\|^2)\lambda + y_l^2\right].$$

Hence, aside from $(n_B - 2)$ eigenvalues equal to 1, the biggest and the smallest eigenvalues are derived from the solution of the second order equation

$$\lambda = \frac{(1 + \|y\|^2) \pm \sqrt{(1 + \|y\|^2)^2 - 4y_l^2}}{2}$$

therefore, the condition number of the matrix η is

$$cond(\eta) = \frac{\sqrt{(1+\|y\|^2) + \sqrt{(1+\|y\|^2)^2 - 4y_l^2}}}{\sqrt{(1+\|y\|^2) - \sqrt{(1+\|y\|^2)^2 - 4y_l^2}}}$$

Since we are interested in finding the matrix η with the lowest condition number among those corresponding to the superbasic columns that are candidate to enter the basic set, we may as well consider the variable corresponding to the column with the lowest condition number:

$$cond(\eta) = \frac{2y_l}{(1+\|y\|^2) - \sqrt{(1+\|y\|^2)^2 - 4y_l^2}}$$

The condition number should be computed for every candidate column.

In practice we have employed a much simpler rule based on the value of y_l (the only eigenvalue different from 1 of the η matrix): choose the entering column with the smallest value either y_l, for $y_l > 1$, or $\frac{1}{y_l}$, for $y_l < 1$. This rule produced satisfactory accuracy.

3.3.8 Initial point

The M&S algorithm requires an initial feasible point (that is a basis) which therefore means finding a set of variables and its relative columns (and vertices) in which the constraints are satisfied. A more detailed explanation of this procedure can be found in [50].

We start with a small basis, $M_B^{(0)} \in \mathbb{R}^{n_i \times n_i}$, where only the convexity constraints are feasible. Then, a maximization problem is solved for each infeasible non-LMC (given that they are a greater-or-equal constraints) until the constraint is satisfied. Next, the basis increases in size with one more column and row. The feasibility of the constraints already included is maintained for successive problems.

Phase I

If there were no non-LMCs, any set of vertices would be feasible. For convenience, the chosen vertices (one for each interval) are those that result from the cost merit order, and the related variable λ will be 1. Therefore, the initial basis, $M_B^{(0)} \in \mathbb{R}^{n_i \times n_i}$, only satisfies the convexity constraints:

$$M_B^{(0)} = [\ I\ ]$$

Phase II

Suppose constraint k is not feasible, that is

$$A_k V_B \lambda < a_k$$

where A_k is the kth row of the non-LMC matrix and a_k its rhs, V_B are the current basic vertices. Suppose also that the basis $M_B^{(k-1)}$ is feasible for the $A^{(k-1)}$ former constraints.

The problem

$$
\begin{aligned}
\underset{\lambda}{\text{maximize}}\ & A_k' V \lambda \\
\text{subject to}\ & A^{(k-1)} V \lambda - I\sigma = a^{(k-1)} \\
& e\lambda^i = 1 \qquad \forall i \\
& \lambda \geq 0,\ \sigma \geq 0
\end{aligned} \tag{3.19}
$$

is solved with the M&S procedure, starting with the last solution point. The algorithm is stopped when the constraint is already feasible. The outline of the method is as follows:

Initial point procedure

i Compute an initial basis , which only considers the convexity constraints (Phase I).

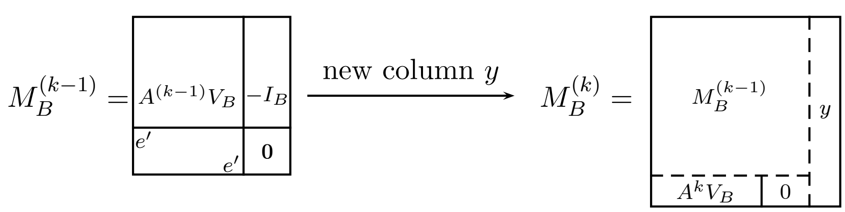
ii Compute the multipliers of the active constraints.

iii Price the nonbasic variables and choose a candidate. Again, the candidates are the variables corresponding to a vertex (from any interval) or a surplus, which are computed similarly to (3.11) and (3.12).

iv Compute the maximum step allowed by the basic variables: α_B.

v Compute the minimum step in order for the constraint k to be feasible: α_k.

vi If $\alpha_k \leq \alpha_B$ then the column corresponding to the candidate variable, y, extends the basis as,



otherwise, the basis is updated (the variable that limits the full α_k step is substituted by the new candidate) and the algorithm proceeds from (ii).

Note that there are no superbasic variables in the initial procedure. This is because problem (3.19) is linear and in this case the M&S algorithm is equivalent to the Simplex.

If constraint k is already feasible given the current basic variables, its surplus is the y column that extends the basis.

3.3.9 Management of the nonbasic set

The main advantage of the column-generation procedure is that vertices (columns) are only generated when they are required. The basic and superbasic vertices have to be generated and stored. Initially there is no nonbasic vertex but as the procedure evolves some vertices become nonbasic. At this point there are two options: discard them or store them and in a following iteration the stored nonbasic vertices are priced and any stored nonbasic vertex may become superbasic.

The version in which nonbasic vertices are deleted is called DWa. This version coincides with the one developed by Ford and Fulkerson [20] for a multi-commodity network problem. The version in which the generated nonbasic vertices are kept is called DWb. In the DWb algorithm, before generating a new vertex the multipliers of the stored nonbasic vertices are computed and if any are negative, they are re-entered as a superbasic.

It makes sense to use the DWb algorithm when the generation of vertices may prove expensive (which in our case is highly related to the data of the problem, for example, the peak load of each interval). The trade-off is the pricing of the stored vertices, which may develop into a long list. The implementation limits the number of stored vertices to 500, and when the list is full the initial vertices are replaced.

3.4 Computational Results

3.4.1 Comparison of the two implementations of the Dantzig-Wolfe algorithm

Either version of the algorithm DWa and DWb achieves the same optimizer (see table 3.1). Comparison of the performance between the two implementations leads us to the following conclusions:

- The version that stores non-basic vertices, DWb, carries out more iterations.
- Surprisingly, the DWb version generates more vertices than the DWa version in 7 out of the 19 cases solved.
- As can be seen in the last column, the same vertex can be reused more than once.
- In 14 out of the 19 test cases the DWa version is faster than the DWb version.

The overall conclusion is that the implementation of the Dantzig-Wolfe procedure that discards the non-basic vertices (DWa) is slightly better and has a simpler formulation. However, both versions show a slow convergence near the optimum.

3.4.2 Performance of the Dantzig-Wolfe algorithm compared to the active set method

Another methodology that has been proposed [7, 44] for solving a problem using the Bloom and Gallant formulation is the active set method [25]. An active set method for the LTGP problem also starts from a feasible point and uses an oracle proposed by Bloom and Gallant to find a new LMC that replaces an LMC that leaves the active set.

The active set and Dantzig-Wolfe column-generation methodologies were tested with the first four cases described in appendix A.1, but using a linear objective function which minimizes the generation

		Objective Function	time (sec)	Total ite	n_S	Vertex gen.	Vertex reused
ltp_01	DWa	9536489725.22	1	250	52	204	-
	DWb	9536489725.22	2	280	52	222	45
ltp_02	DWa	10961049198.47	6	841	89	599	-
	DWb	10961049198.42	10	1868	89	560	490
ltp_03	DWa	10977720292.76	14	1341	106	700	-
	DWb	10977720295.11	16	1966	105	664	363
ltp_04	DWa	10979064717.68	15	1645	106	727	-
	DWb	10979064725.51	30	3198	106	789	754
ltp_05	DWa	8840160418.74	90	2761	314	1179	-
	DWb	8840160418.75	183	4906	314	1375	1062
ltp_06	DWa	10667293264.28	336	13386	321	1861	-
	DWb	10667012493.53	690	23264	320	1745	2686
ltp_07	DWa	5862769730.45	6467	8702	502	3901	-
	DWb	5862769732.06	12365	17572	502	8794	4517
ltp_08	DWa	7077804321.42	1635	4170	444	1914	-
	DWb	7077804321.55	4025	10294	444	2298	3148
ltp_09	DWa	5759806317.36	6805	5604	640	8412	-
	DWb	5759806316.07	5920	6087	648	3220	1690
ltp_10	DWa	5267939456.14	69	2175	143	1576	-
	DWb	5267939445.56	81	2721	143	934	689
ltp_11	DWa	4868047074.81	312	1295	379	2196	-
	DWb	4868047082.93	285	1455	379	941	293
ltp_12	DWa	5122060084.62	8954	48547	396	16301	-
	DWb	5122060082.95	12122	72345	396	11506	19289
ltp_13	DWa	6623277319.17	735879	5100	2585	6144	-
	DWb	6623277319.16	519125	6806	2585	5819	1126
ltp_14	DWa	4970958818.52	11	573	158	510	-
	DWb	4970958818.52	13	727	158	560	117
ltp_15	DWa	4740125611.15	44	692	273	873	-
	DWb	4740125635.57	55	899	273	797	134
ltp_16	DWa	6022480775.29	29	528	260	763	-
	DWb	6022480775.29	31	608	260	646	74
ltp_17	DWa	7050449319.63	92	841	316	1222	-
	DWb	7050449319.63	78	1094	316	781	200
ltp_18	DWa	4701104749.99	188	1071	334	1138	-
	DWb	4701104755.56	211	1247	334	1072	165
ltp_19	DWa	6999581717.45	1750	1443	674	1570	-
	DWb	6999581718.20	1662	1655	674	1601	194

Table 3.1: Comparison of the solutions found using the Dantzig-Wolfe algorithm, a & b versions

costs (therefore the problem name has a ”b” appended). These cases were run on a SPECfp2000 310 processor in a Hewlet Packard Netserver LC2000 U3.

Both the active set and the Dantzig-Wolfe methods require a considerable number of iterations to reach a feasible solution. The number of iterations appears under the heading *feas. iters.* (feasibility iterations) in table 3.2, after which the number of iterations needed to achieve the optimizer is shown. The next column contains the required CPU time. The last column (*dig. ag.*) shows the number of agreement figures of the optimal objective function found with the two methodologies.

	active set method			Dantzig-Wolfe cg			
	feas. iters.	total iters.	time (s)	feas. iters.	total iters.	time (s)	dig. ag.
ltp_01b	239	312	9.0	21	224	16.4	9
ltp_02b	513	734	80.1	128	516	16.1	10
ltp_03b	781	1096	348.0	310	1213	21.6	9
ltp_04b	1075	1404	756.6	400	1768	38.5	9

Table 3.2: Comparison of the active set and the Dantzig-Wolfe column-generation method

Several conclusions can be drawn from the results of table 3.2. The first is that the column-generation method is quicker to reach the optimal solution and that the rate of increase of time required with problem size is lower in the case of Dantzig-Wolfe than when using the active set solution. Both methodologies reach practically the same optimizer.

3.4.3 Convergence

The column-generation algorithm shows slow convergence near the optimum. Figure 3.1 shows the gap (in log (base 10) scale) of the objective value (DWa version) against the time. The distance between two solid circles is the time required by the algorithm to carry out 100 iterations.

The Murtagh and Saunders algorithm finishes when any of the nonbasic variables reaches a sufficiently negative price. In practice, it stops when none of the vertices found (one for each interval) and none of the nonbasic slacks have a Lagrange multiplier, ν, smaller than $-10^{-12}\|\pi\|$.

The objective gap is computed after the iterations have finished to show the convergence of the algorithm. The main reason for such a slow convergence is that the solution points are a combination of several vertices, which makes accuracy harder to achieve.

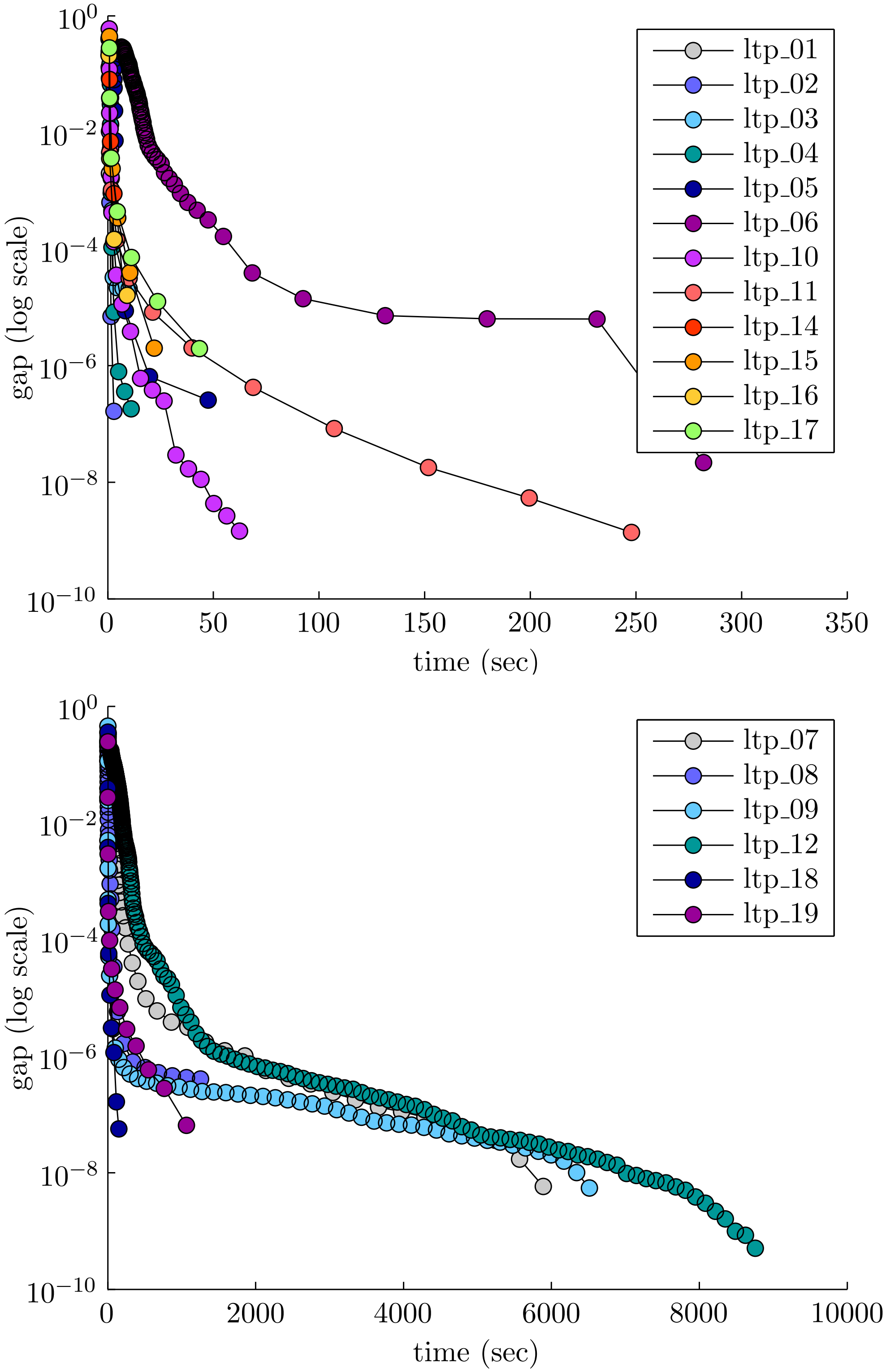


Figure 3.1: Convergence of the objective function to the optimizer (gap between the objective value and the optimal value in log scale versus time)

Chapter 4

Direct solution methods and heuristic procedure

There are several procedures for solving a problem such as the long-term generation planning (2.24). The active set method is one that considers subsets of active load-matching constraints (2.24b) with, at most, $n_i \cdot n_u$ active linear inequalities (as many as there are variables), and discards constraints from this set and enters new ones as the optimization proceeds [44]. There is an *oracle* associated with load-matching inequalities in the Bloom and Gallant formulation [7], 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.

Column generation methods such as the Ford-Fulkerson and Dantzig-Wolfe methods find feasible points and the solution by determining a convex combination of the vertices of the polyhedron defined by all the LMCs [46]. When one uses the Bloom and Gallant formulation, each of these vertices corresponds to a perfect ordering (as described in 2.4.1) of a given order, which is determined by certain modified costs.

In both active-set or column-generation methodologies there is no need to explicitly create the exponential number of LMCs (with their rhs that take a long time to compute).

However, should we attempt to employ direct quadratic programming [25] or interior-point quadratic programming [69], we would explicitly require the exponential number of inequalities, with their exponential number of slack variables and Lagrange multipliers. In practice this would render the solution for $n_u \geq 20$ impossible. It is clear, however, that there are only a reduced number of active LMCs at the optimizer, and that finding this optimal active set is as difficult as finding the solution.

In this chapter a heuristic is put forward for building the optimal active set of inequality LMCs. This heuristic employs a reduced subset of LMCs and is moderately enlarged in successive steps until the optimal active set and solution are found.

4.1 Heuristic to determine a loading order

The heuristic we present herein solves the LTGP problem for a fixed maintenance schedule and will be referred to as GP heuristic [51]. It builds the solution iteratively along with the loading order. The GP heuristic is based on the fact that each solution corresponds to a loading order, and the active LMCs at the optimizer will be nested [42]. It is an iterative process in which a few LMCs are added in successive steps.

4.1.1 Notation

The heuristic consists in solving problem (2.24) several times, although a different subset of LMCs is used each time instead of the complete set.

Let L^i be the list of LMCs used in interval i. Each element of L^i is a set of units which defines a unique LMC. Matrix $B^i_{L^i}$ represents the submatrix of LMCs defined in L^i, and $r^i_{L^i}$ its rhs.

Example

Suppose a case with 5 units. If the list L^i has two elements: $\{\{3,5,2\},\Omega\}$, $B^i_{L^i}x^i \leq r^i_{L^i}$ stands for:

$$B^i_{L^i}x^i = \begin{bmatrix} B^i_{\{3,5,2\}} \\ B^i_{\Omega} \end{bmatrix} x^i = \begin{bmatrix} . & 1 & 1 & . & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} x^i \leq r^i_{L^i} = \begin{bmatrix} r^i_{\{3,5,2\}} \\ r^i_{\Omega} \end{bmatrix}$$

At each iteration the list is updated and a relaxation (in terms of LMCs) of problem (2.24) is solved:

$$\begin{aligned} &\underset{x}{\text{minimize}}\ h'x + \frac{1}{2}x'Hx \\ &\text{subject to}\ Ax \geq a \\ &\qquad B^i_{L^i}x^i \leq r^i_{L^i} && \forall\, i \\ &\qquad 0 \leq x^i_j \leq \overline{x}^i_j && \forall\, i \quad \forall j \in \Phi \end{aligned} \tag{4.1}$$

Note that in problem (4.1) the upper bound of the variables is considered explicitly ($\overline{x}^i_j = r^i_{\{j\}} = t^i(1-q_j)\int_0^{c_j} S^i_{\varnothing}(z)dz$).

4.1.2 Steps of the heuristic

LMCs tailor the shape of the generation duration curve by limiting the maximum generation of the units in accordance with the loading order. Candidate constraints to be violated are the ones whose units are closer to its upper bound. We define the ratio between the expected generation and its upper bound as

$$\rho^i_j := \frac{x^i_j}{\overline{x}^i_j} \quad \forall\, i \quad \forall\, j \tag{4.2}$$

The heuristic builds a loading order for each interval. However, in practice we only need to keep track of the units already ordered, because these have to be included in the following LMCs added to the problem, so as to nest the other constraints of the interval. Let ϕ^i be the set of units already ordered in interval i.

The heuristic proceeds following three different rules depending on the information available:

- Initially, a solution with only the all-one LMC is found.
- From this solution we select the units at upper bound from each interval ($\rho^i_j \simeq 1$) and these will be the first units in the loading order. Usually, this is a small subset in each interval. These units compete for the top positions. We do not decide their ordering, so the list L^i is increased by any LMC made up only of units at upper bound.
- From then on, a unit from each interval is chosen and a new solution is found with the new LMC. The unit and interval next in order is the one with highest ρ^i_j value. This choice answers the question of which LMC is more likely to be violated.

GP heuristic

i Initialization

- $\phi^i := \varnothing \quad \forall\, i$
- $L^i := \{\Omega^i\} \quad \forall\, i$
- Solve (4.1)

ii Self-ordering

- $\phi^i := \{j \in \Omega^i \,|\, \rho^i_j \simeq 1\}$
- $L^i := L^i \cup \{\omega \,|\, \forall\, \omega \subseteq \phi^i\} \quad \forall\, i$
- Solve (4.1)

iii While $|\Omega^i \setminus \phi^i| > 1 \quad \forall\, i$, order step-by-step:

- $\phi^i := \phi^i \cup \{j \,|\, \rho^i_j = \max_{\forall\, k \in \Omega^i \setminus \phi^i} \rho^i_k\} \quad \forall\, i$
- $L^i := L^i \cup \{\phi^i\} \quad \forall\, i$
- Solve (4.1)

The heuristic stops at the point where all units of all intervals are ordered. If one interval forms the loading order before the iterations end, no further constraints are added to that interval.

Comments on the heuristic

For the expected energies x^i_j, the first solution to (4.1) in the heuristic only considers the upper bounds and the all-one LMCs, but not the rest of the LMCs. The non-load-matching constraints and the objective function will steer a subset of the units to their upper bound. The heuristic then includes all possible LMCs regarding this subset in the problem, which reduces the expected generation of these units, so that they conform to the shape of the GDC for the LDC of each interval. Other units then increase their expected energy generation, and they will be made to conform, one by one, to the GDC by nesting whichever one is closest to its upper bound with those units that are already ordered. In view of the fact that every time a constraint or a group of constraints is added the optimization is solved with all the non-LMCs, the solution, whose units are all nested, will most probably be feasible and optimal.

As the heuristic ends when all the units are nested (and one of the as yet unnested units is included per iteration), there will be at most n_u iterations of the heuristic.

In general not all the constraints added by the heuristic will be active at the optimizer, as the solution point will most probably have a partial ordering.

As regards implementation, a safeguard to limit the maximum number of units at upper bound in the self-ordering part is necessary. For the test cases solved, the number of units at upper bound never exceeded ten units (per interval), which means that the number of LMCs to be added (at most $(2^{10}-1)\cdot n_i = 1023n_i$) is acceptable. Therefore, the application of the heuristic is limited, in theory, to cases where, as a result of the self-ordering part (ii), there are not all that many units (for example, 18) whose expected energy is at its upper limit, which are the most efficient ones. However, in practice the limitation may be the number of LMCs included in stage (ii), and this may allow a higher number of units at upper bound because there is no need to include the exponential number of LMCs, given that the constraints corresponding to a subset ω such that $\sum_{j\in\omega} c_j \leq \underline{p}$, where $\underline{p}$ is the base power of the LDC, are linear combinations of the upper bounds of the expected energies, as can be easily deduced from the expressions (2.2) and (2.1). Moreover, as is common in 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 units but with equal overall efficiency.

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

4.2 Checking the feasibility of a solution found through the GP heuristic

There is no direct methodology to check whether the solution is feasible or not. However, we can characterize some situations in order to avoid checking all the LMCs. In this section we will apply the following properties:

- $S_\zeta(z) \geq S_\theta(z) \quad \forall\, \zeta \subseteq \theta$ (see section 2.2.4)
- The rhs of an LMC defined by the set χ can be computed as

$$r_\chi := \sum_{j\in\chi} x_{j|\chi_{o_j}} = t\sum_{j\in\chi}(1-q_j)\int_0^{c_j} S_{\chi_{o_j}}(z)dz \tag{4.3}$$

(see equations (2.6) and (2.1)). Any ordering of the units, χ_o, is valid for computing the rhs (see section 2.4.1).

Without loss of generality, we 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.

Example

Suppose a case with 5 units with a solution point following a loading order $\Omega_o = \{3, 5, 2, 1, 4\}$. To compute an LMC defined by the subset χ which has units u_2, u_3 and u_4 we can use loading order $\chi_o = \{3, 2, 4\}$, which is based on Ω_o.

The loading order $\Omega_o = \{3, 5, 2, 1, 4\}$ means that the first unit to be loaded is u_3, the second is u_5, the third is u_2, the fourth is u_1 and the last one is u_4. Therefore, the right-hand side of the LMC defined by χ can be computed loading first u_3, second u_2 and finally u_4.

In this section, given that the LMCs are defined for each LDC, we will refer to a single interval although the results apply to each interval.

4.2.1 Characteristics of the active load-matching constraints of a feasible point

The structure of the coefficients on the left-hand side of any LMC (2.24b) is a row vector of ones and zeros, depending on which units are present in the subset ω under consideration. Regarding these ones and zeros that make up the active LMCs, the ones must be nested whatever the solution point.

A solution point with a perfect ordering has n_u active LMCs. The active constraints can be ordered in such a way that each subsequent LMC has one more 1 than the preceding constraint. Units can be reordered to produce a structure of ones shaped like a flight of stairs.

Example

Suppose a solution of a case with 5 units with loading order $\Omega_o = \{3,5,2,1,4\}$. The matrix of active LMCs is

$$\begin{bmatrix} B_{\{3\}} \\ B_{\{3,5\}} \\ B_{\{2,3,5\}} \\ B_{\{1,2,3,5\}} \\ B_{\Omega} \end{bmatrix} x = \begin{bmatrix} . & . & 1 & . & . \\ . & . & 1 & . & 1 \\ . & 1 & 1 & . & 1 \\ 1 & 1 & 1 & . & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} =$$

$$\begin{bmatrix} 1 & . & . & . & . \\ 1 & 1 & . & . & . \\ 1 & 1 & 1 & . & . \\ 1 & 1 & 1 & 1 & . \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_3 \\ x_5 \\ x_2 \\ x_1 \\ x_4 \end{bmatrix} = \begin{bmatrix} r_{\{3\}} \\ r_{\{3,5\}} \\ r_{\{2,3,5\}} \\ r_{\{1,2,3,5\}} \\ r_{\Omega} \end{bmatrix}$$

where the matrix with the columns arranged following the loading order shows the flight-of-stairs structure.

A solution point with partial ordering has less than n_u LMCs active. When the difference in the number of units between two successive nested active constraints is greater than 1, there is said to be a *landing*.

Example

Suppose a solution of a case with 5 units with loading order $\Omega_o = \{3,(5,2,1),4\}$. The matrix of active LMCs is

$$\begin{bmatrix} B_{\{3\}} \\ B_{\{1,2,3,5\}} \\ B_{\Omega} \end{bmatrix} x = \begin{bmatrix} . & . & 1 & . & . \\ 1 & 1 & 1 & . & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 1 & . & . & . & . \\ 1 & 1 & 1 & 1 & . \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_3 \\ x_5 \\ x_2 \\ x_1 \\ x_4 \end{bmatrix} =$$

$$\begin{bmatrix} r_{\{3\}} \\ r_{\{1,2,3,5\}} \\ r_{\Omega} \end{bmatrix}$$

where the matrix with the columns arranged following the loading order shows the landing in the stair structure between constraints $B_{\{3\}}$ and $B_{\{1,2,3,5\}}$.

4.2.2 Feasibility with a perfect ordering entails complete feasibility

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

Lemma 4.2.1. *A solution with perfect ordering is feasible.*

Proof. To check the feasibility of a perfect ordering solution with order Ω_o, we can consider any other (inactive) LMC χ that only involves a subset of units of Ω. Using (4.3), the alternative LMC

$$\sum_{j\in\chi} x_{j|\Omega_j} = t\sum_{j\in\chi}(1-q_j)\int_0^{c_j} S_{\Omega_{o_j}}(z)dz \leq r_\chi = t\sum_{j\in\chi}(1-q_j)\int_0^{c_j} S_{\chi_{o_j}}(z)dz$$

is also satisfied as $S_{\Omega_{o_j}}(z) \leq S_{\chi_{o_j}}(z)\ \forall\, z$, given that, for the same loading order, for any unit j, the units already loaded in the case of subset χ_{o_j} will be the same or less than in the case of using subset Ω_{o_j}. Therefore, a solution with a perfect ordering is feasible. □

4.2.3 Potentially violated constraints 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.

Lemma 4.2.2. *In a partial ordering solution, the only load-matching constraints that might be violated correspond to constraints made up of all the units ordered prior to the landing plus the units that form the landing.*

Proof. The perfectly ordered units correspond to a set of nested and active LMCs. By subtraction of successive nested constraints, the expected value of a perfectly ordered unit is found and corresponds to expression (2.1). As shown in lemma (4.2.1), the expected generation of these units is feasible for any LMC.

When a subset of units does not have a perfect ordering, which means that the position of at least one unit is split by others in the subset, there is nothing to prevent an ordering of these units from being infeasible. The ordering of these units takes into account the set of units that has already been loaded. □

In practice, this means that one has to check the LMCs of all the combinations of the units that are not perfectly ordered. The combinations are exponential, $2^{n_l}-2$, being n_l the number of not perfectly ordered units (the split unit plus the splitting one(s)).

Example

Suppose a case with loading order $\Omega_o = \{3,(5,2,1),4\}$. The potentially violated LMCs are:

$$\begin{bmatrix} B_{\{3,1\}} \\ B_{\{3,2\}} \\ B_{\{3,5\}} \\ B_{\{3,1,2\}} \\ B_{\{3,1,5\}} \\ B_{\{3,2,5\}} \end{bmatrix} x = \begin{bmatrix} 1 & . & 1 & . & . \\ . & 1 & 1 & . & . \\ . & . & 1 & . & 1 \\ 1 & 1 & 1 & . & . \\ 1 & . & 1 & . & 1 \\ . & 1 & 1 & . & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \geqslant \begin{bmatrix} r_{\{3,1\}} \\ r_{\{3,2\}} \\ r_{\{3,5\}} \\ r_{\{3,1,2\}} \\ r_{\{3,1,5\}} \\ r_{\{3,2,5\}} \end{bmatrix}$$

Note that unit 3, which is ordered prior to the landing, is always present.

The heuristic does not perform these feasibility checks. They were performed using separate programs to verify the results given by the heuristic.

4.3 Implementation of the GP heuristic for the LTGP problem

The implementation of the GP heuristic requires the definition and solution of various instances of problem (4.1), each with an increasing number of LMCs in the inequalities $B^i_{L_i} x^i \leq r^i_{L_i}$. This can be accomplished in several ways.

4.3.1 Using a modeling language and a general linearly constrained optimization solver

Through a modeling language, it is possible to define the successive problems of type (4.1) to be solved, including the calculation of the rhs values $r^i_{L_i}$ for each LMC under consideration. This was done using the modeling language AMPL [22] and by developing a *script* that codes the algorithm of the heuristic.

In order to solve each problem of type (4.1) the modeling language employed (AMPL) invokes a solver for finding the solution. The favored linearly constrained general solver is *CPLEX 9.1* [16] as this solver is efficient for quadratic programming as problem (4.1) is. Other solvers as *Minos* [41] or *SNOPT* [24] could have been employed.

Given that the calculation of the rhs values $r^i_{L_i}$ for each LMC is very time-consuming, two possibilities arise:

a) the calculation and storage of all possible rhs of the LMCs, and then using the ones that are needed, or

b) the calculation of only the required rhs of the LMCs.

Procedure a) allows the calculation to be done beforehand, but this is only possible in very small cases, because in medium and large cases the calculus would take up a lot of CPU time and storing the rhs values would require an enormous amount of memory. Procedure b) can be implemented in two ways:

b1) by computing the rhs of the requested LMCs in the modeling language employed within the script developed for the heuristic, which is not efficient for heavy arithmetic calculations, or

b2) by computing the rhs of the requested LMCs through an external routine in a calculus-efficient language (such as C++) and linking this external routine to the modeling language with a *hook* (a code specially developed to that purpose) [1]

Procedures a), b1) and b2) were all tried with *AMPL* and *CPLEX*.

[1] The contribution of my colleague Matteo Tesser in implementing the C-AMPL hook is here gratefully acknowledged.

The computational results reported in table 4.1 show that implementation procedure b2) is quite efficient and compares favorably with the solution to the same problems obtained column-generation methods.

	LMCs rhs comp.		Solution time		
	time	size	read (a)	AMPL (b1)	hook C++ (b2)
ltp_01	59	4M	0.45	201.97	1.10
ltp_02	257	15M	1.18	226.81	1.04
ltp_03	1113	57M	3.59	284.97	1.16
ltp_04	2313	113M	5.87	324.96	1.25
ltp_14	139	9M	0.78	403.70	1.05
ltp_15	300	18M	1.29	540.50	1.20
ltp_16	77	4M	0.85	524.78	1.11
ltp_17	1307	70M	4.94	688.90	1.58
ltp_18	6077	277M	13.63	852.61	2.42

Table 4.1: Time (in seconds) required to compute all the LMC rhs and to solve the cases using the heuristic, either by reading the rhs or computing them in AMPL or C++.

4.3.2 Using a special purpose code and a special purpose linearly constrained optimization solver

Both the heuristic, with its definition of problems of type (4.1), and a special purpose solver implementing an interior-point method for quadratic programming were also developed in C language. In chapters 5 and 6 of this thesis, the details of the interior point code and the computational results obtained are reported.

4.4 Computational results

Table 4.2 shows the results obtained using the GP heuristic to solve the test cases explained in appendix A.1. The first column shows the objective function, which is a value comparable to that obtained using a column generation method (see table 3.1). The next column shows the time in seconds used to solve each instance with the heuristic coded in AMPL, using CPLEX as a solver and hooking a routine coded in C to AMPL in order to compute the required LMC rhs. The total time is only a few seconds and the greater part of this time is used to compute the LMC rhs.

The following columns give some statistics on the heuristic. The column headed *Ite* shows the number of problems solved by the GP heuristic (the problem is resolved only when the LMCs added on are infeasible). The cases with the number of iterations equal to 1 (like cases ltp_12 or ltp_13) did not solve any problems in the step-by-step stage and none of the constraints derived in the self-ordering stage were infeasible. The next two columns display, at the optimizer, the number of active LMCs and the number of units at upper bound (not counted as LMCs). The total numbers of computed LMCs are shown alongside.

	Obj. Func.	t (s)	Ite	Optimizer act. LMCs	act. bo.	comp. LMCs	Self-ord. units u.b.	num. LMCs
ltp_01	9536489768.18	1.12	5	89	59	179	59	106
ltp_02	10961049201.77	1.15	8	37	22	154	22	22
ltp_03	10977720301.34	1.22	10	39	23	175	23	22
ltp_04	10979064730.92	1.24	11	42	24	185	24	22
ltp_05	8840160481.31	4.30	2	1065	108	2370	108	1994
ltp_06	10667293273.73	3.66	2	318	82	2594	82	1994
ltp_07	5862769740.45	1.78	7	126	69	867	69	52
ltp_08	7077804321.59	2.10	12	89	76	620	76	48
ltp_09	5759806317.45	2.58	6	26	32	738	42	15
ltp_10	5267939454.75	1.06	9	24	9	223	14	13
ltp_11	4868047083.60	1.44	4	14	12	407	22	13
ltp_12	5122061425.37	2.91	1	241	74	931	74	15
ltp_13	6623277320.95	7.31	1	87	218	2675	237	52
ltp_14	4970958829.89	1.01	6	33	127	224	127	27
ltp_15	4740125882.48	1.22	6	34	40	326	52	27
ltp_16	6022480779.55	1.11	5	28	15	290	34	27
ltp_17	7050449326.05	1.51	11	37	37	384	48	27
ltp_18	4701104763.65	2.15	13	82	42	434	52	27
ltp_19	6999581723.57	3.23	14	77	112	806	112	27
avg		*2.22*	*7*	*131*	*62*	*767*	*67*	*238*

Table 4.2: Solution obtained with the GP heuristic using AMPL, the CPLEX barrier QP solver and a hook to a code in C for calculating the LMC right-hand sides

The last columns of the table present results from the initialization and self-ordering stage of the GP heuristic. They show the number of units at upper bound when solving the problem with only the all-one LMC at each interval, and the number of LMCs generated in the self-ordering part (using only the LMCs whose sum of the unit capacities that define it is larger than the base load). It can be seen that the number of LMCs is far from being exponential to the number of units at upper bound.

4.5 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 [14] and also used by Bloom and Gallant [7]. 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. It is a single-interval problem ($n_i = 1$) with a duration of $t = 8$. Units must match the LDC shown in figure 4.1. The data for the problem, available at [32] in AMPL format, is shown in table 4.3. The maximum generation of each unit ($t(1 - q_j)c_j$) is displayed in table 4.4.

	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{ext}
c_j	2	0.4286	2.5714	1	1	1	1	1	1	∞ (100)
f_j	2	10	12	13	15.5	0	0	0	0	20
q_j	0	0	0	0	0	0	0	0	0	0

Table 4.3: Small case data

	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
$\overline{x}_j$	16	3.43	20.57	8	8	8	8	8	8

Table 4.4: Generation upper bound.

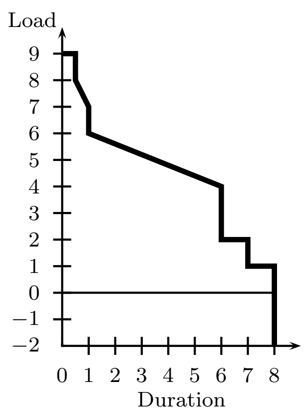


Figure 4.1: Load duration curve

Two linear equality non-load matching constraints are used to represent the storage units:

$$\begin{aligned} 0.7\,x_6 + x_7 &= 5.6 \\ 0.75\,x_8 + x_9 &= 6.0 \end{aligned}$$

GP heuristic

The sequence of the list of LMCs considered in each iteration of the GP heuristic is shown in table 4.7. The solution of the expected generation of the units at each iteration is shown in table 4.5. Table 4.6 shows the ratio between the expected generation solution, x_j, and its upper bound, $\overline{x}_j^i$, in each iteration of the heuristic.

- i **Initialization stage**

  First, solve the problem with only the all-one LMC and the unit upper bounds.

- ii **Self-ordering stage**

  From the solution of the initialization part (see table 4.5), the set of units generating at maximum capacity ($\rho \simeq 1$) is $\{u_1, u_2, u_6, u_8\}$ (see table 4.6). All possible LMCs formed by units at maximum capacity are considered in the next problem solved. Note that the single unit LMC is already in the problem, expressed as the upper bound of the variable.

- iii **Step-by-step stage**

  Subsequent LMCs must nest units in $\phi = \{u_1, u_2, u_6, u_8\}$. Therefore, the units to be ordered are $\Omega \backslash \phi := \{u_3, u_4, u_5, u_7, u_9\}$. As 4 units out of 9 have been ordered, there will be 4 more iterations (there is no choice of position for the last unit ordered).

  - Among the units as yet unordered, the nearest to its upper bound is u_3 with a ratio of 0.87, which leads us to consider the constraint: $\{u_1, u_2, u_3, u_6, u_8\}$. The set of units ordered ϕ and the list L are updated accordingly (see table 4.7).
  - Unit u_4, with $\rho_4 = 0.47$, is the unordered unit with the highest value.
  - We repeat the procedure with unit u_9 as it is the unordered unit with the highest ρ value (0.13).
  - Finally, the last constraint to be added to the model is the one that has u_5 (with ρ_5 equal to 0.11), which gives the optimal solution.

Note that u_7 is still unordered but if we carry out another iteration it would give the all-one LMC equation that is already in the model. On observing the active LMCs of the optimal solution, we can deduce the loading order: $\Omega_o = \{1, 2, (6, 8), 3, 4, (9, 5), 7\}$ (see section 2.4.1).

x_j	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
i	16	3.43	16.83	0	0	8	0	8	0
ii	16	2.57	17.93	0	0	8	0	7	0.75
iii	16	2.57	14.18	3.75	0	8	0	7	0.75
	16	3.43	14.18	2.25	0.88	7.57	0.30	6.57	1.07
	16	3.43	14.18	2.25	0.88	7.48	0.37	6.67	1
*	16	3.43	14.18	2.25	0.89	7.29	0.5	6.86	0.86

Table 4.5: Evolution of the expected generation (x_j) of the units along the three main stages of the GP heuristic.

ρ_j	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
i	**1**	**1**	**0.82**	**0**	**0**	**1**	**0**	**1**	**0**
ii	1	0.75	**0.87**	**0**	**0**	1	**0**	0.875	**0.09**
iii	1	0.75	0.69	**0.47**	**0**	1	**0**	0.875	**0.09**
	1	1	0.69	0.28	**0.11**	0.95	**0.04**	0.82	**0.13**
	1	1	0.69	0.28	**0.11**	0.93	**0.05**	0.83	0.125
*	1	1	0.69	0.28	0.11	0.91	**0.06**	0.86	0.11

Table 4.6: Evolution of the ρ_j value along the three main stages of the GP heuristic. Bold-face ratios correspond to units that are candidate to be ordered.

	ϕ	L
i	$\varnothing$	$\{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9\} \cup$
ii	$\{u_1, u_2, u_6, u_8\}$	$\{\{u_1, u_2\}, \{u_1, u_6\}, \{u_1, u_8\}, \{u_2, u_6\}, \{u_2, u_8\}, \{u_6, u_8\}, \{u_1, u_2, u_6\}, \{u_1, u_2, u_8\}, \{u_1, u_6, u_8\}, \{u_2, u_6, u_8\}, \{u_1, u_2, u_6, u_8\}\} \cup$
iii	$\{u_1, u_2, u_3, u_6, u_8\}$ $\{u_1, u_2, u_3, u_4, u_6, u_8\}$ $\{u_1, u_2, u_3, u_4, u_6, u_8, u_9\}$ $\{u_1, u_2, u_3, u_4, u_5, u_6, u_8, u_9\}$	$\{u_1, u_2, u_3, u_6, u_8\} \cup$ $\{u_1, u_2, u_3, u_4, u_6, u_8\} \cup$ $\{u_1, u_2, u_3, u_4, u_6, u_8, u_9\} \cup$ $\{u_1, u_2, u_3, u_4, u_5, u_6, u_8, u_9\}$

Table 4.7: Set of units ordered (ϕ) in each iteration of the GP heuristic and evolution of the list of considered LMCs.

4.6 Solution to the LTGP with hydro generation influence

The correlation between the hydro generation level and the market price was studied for the last six LTGP problem test cases, presented in appendix A.1 . The LTGP$_{\mathcal{H}}$ problem (introduced in 2.11.2) is the same as the LTGP problem, except that the final market-price function depends on the hydro-generation level. This gives rise to an indefinite matrix in the quadratic objective function and CPLEX is no longer an appropriate solver. Minos is used instead. Cases solved with all the LMCs are:

	Obj. Func.	time (sec)
ltp_wh_14	55725485903.78	58.73
ltp_wh_15	56692993692.60	212.33
ltp_wh_16	56551340123.87	30.37
ltp_wh_17	55589378624.24	3366.1

Table 4.8: Solution of the LTGP$_{\mathcal{H}}$ problem with Minos 5.0 with all the LMCs

Cases ltp_wh_18 and ltp_wh_19 were also prepared, but the exponential number of LMCs exceeded CPU memory. The times indicated in table 4.8 are the solver time, but the computation time of the LMC right-hand sides should also be taken into account (table 4.1).

The GP heuristic was also applied to solve these cases:

	Obj. Func.	time (sec)
ltp_wh_14	55725485903.79	1.21
ltp_wh_15	56692993692.80	1.96
ltp_wh_16	56551340123.73	1.30
ltp_wh_17	55589378624.30	2.38
ltp_wh_18	56317265746.66	2.57
ltp_wh_19	55765847940.70	1.87

Table 4.9: Solution of the LTGP$_{\mathcal{H}}$ problem using the GP heuristic (and using Minos)

The CPU time taken to solve the LTGP$_{\mathcal{H}}$ cases is longer but comparable to the time taken to solve the LTGP ones using the GP heuristic and CPLEX. For cases ltp_wh_14 to ltp_wh_17, the objective function containing all the LMCs or using the reduced subset generated by the GP heuristic is the same. The GP heuristic with indefinite objective function worked well.

Hopefully, the LTGP$_{\mathcal{H}}$ results more closely reflect real market behavior than LTGP results.

4.6.1 Estimation of the new coefficients

The test cases for the LTGP$_{\mathcal{H}}$ problem and for the LTGP problem differ in some respects:

- The fitted market-price function for interval i is:

$$b_0^i - c_0^i \sum_{k \in \mathcal{H}} x_k^i + l^i t,$$

  where $\sum_{k\in\mathcal{H}} x_k^i$ for past intervals is the total hydro generation produced in that period. We estimated the coefficients b_0^i, c_0^i and l^i of the market-price function using data from a period of three years (see figure 4.2).

  In our examples, the set $\mathcal{H}$ is made up of two hydro units, one corresponding to the SGC and another merging the hydro units of the rest of participants.

  In some cases the market-price function in combination with the hydro generation bounds could result in negative profits for durations close to the total interval length. The correlation coefficient c_0^i was modified in order to keep the price-function within some limits, whatever the optimization results.

- The coefficients are valid for the range of data observed. The hydro generation bounds in each interval were slightly widened (see figure 4.3).

- Given that the hydro-generation is penalized by the negative correlation coefficient c_0^i, but in practice the predicted generation will definitely be used, hydro generation is fixed with an equality constraint (instead of the inequality) that balances hydro generation every quarter.

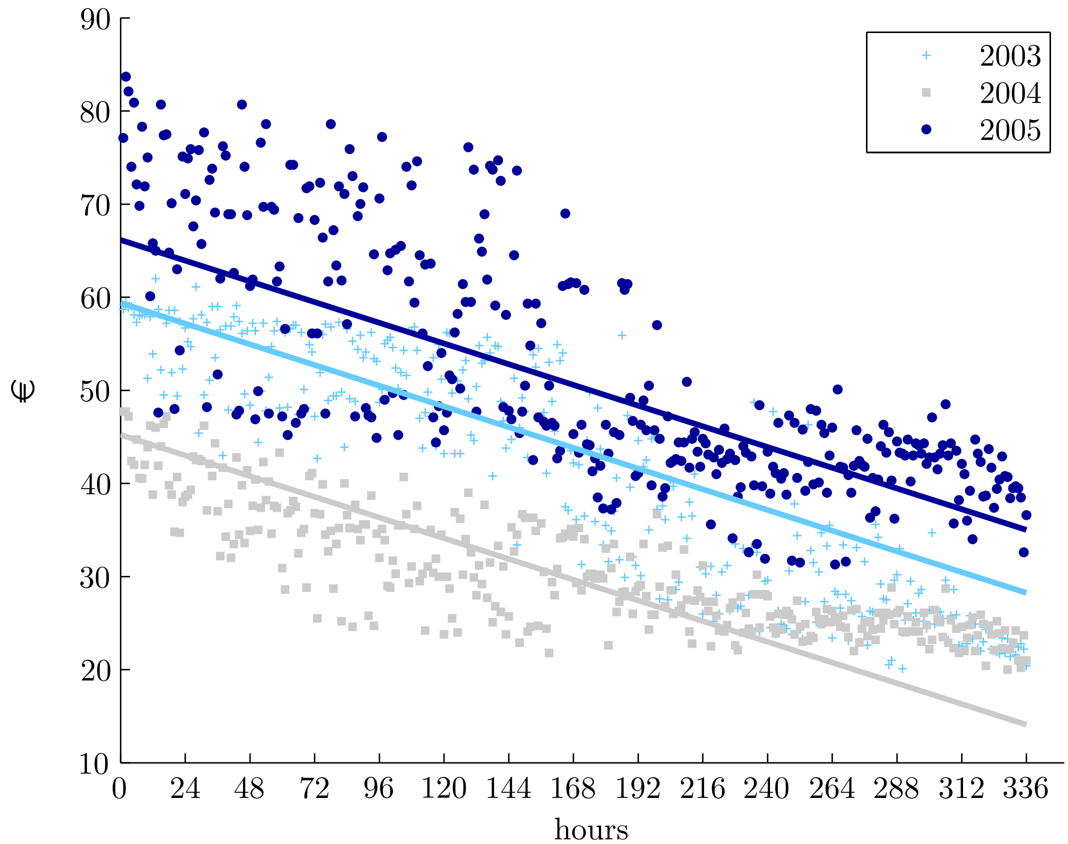


Figure 4.2: Market price ordered by decreasing load and fitted lines in the second interval (15 days) of the case ltp_wh_14 in the years 2003, 2004 and 2005, each having a different level of hydro generation in this interval

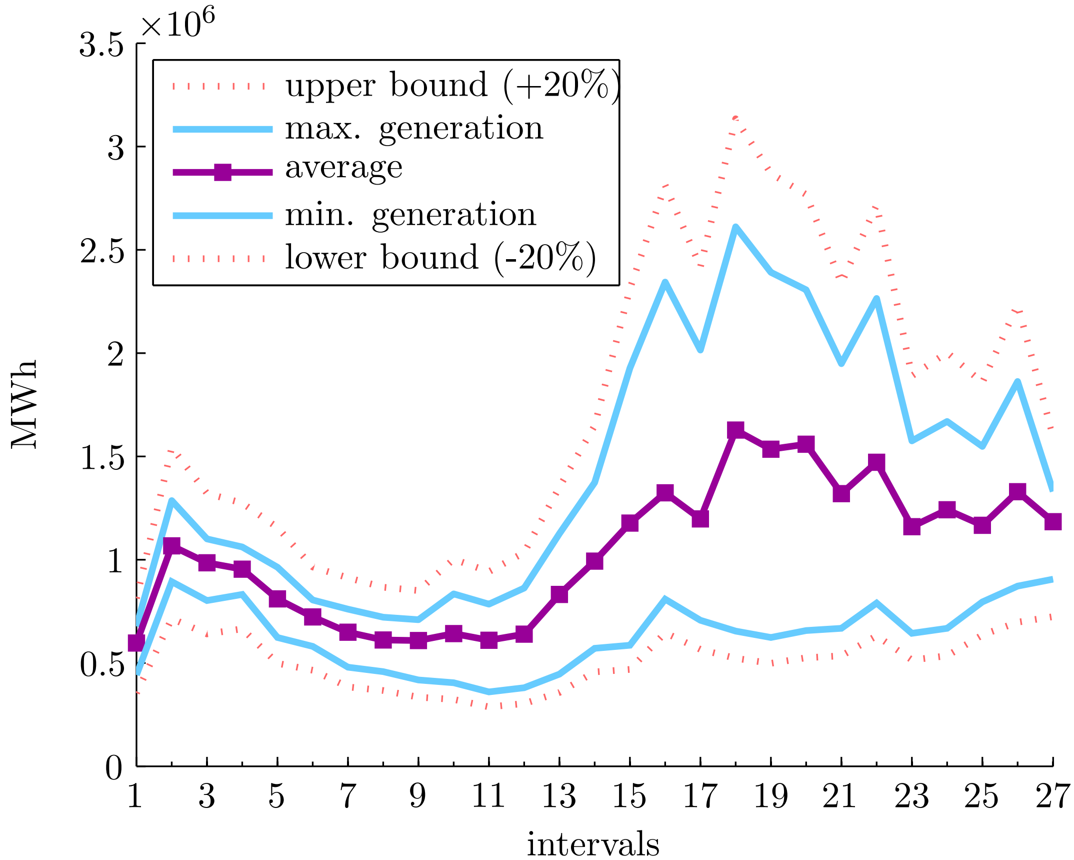


Figure 4.3: Average hydro generation and upper and lower limits in the 27 intervals defined in case ltp_wh_14 for the years 2003, 2004 and 2005

Chapter 5

Application of interior point methods

The heuristic proposed in chapter 4 requires the solution of many similar problems. The problems employ a reduced subset of LMCs that is moderately enlarged in successive steps of the heuristic until the optimal active set and solution are found. The use of direct methods is now appropriate since the number of LMCs to be considered is not large. We solve the problems through an interior point method.

Since the publication of Karmarkar's paper [33] in 1984, interior point methods have evolved both theoretically and computationally being nowadays a well-known field. In this chapter we present the basic steps employed in interior point methods for solving a quadratic problem and our choice of our implementation.

5.1 Infeasible primal-dual interior point method

Primal-dual interior point methods find primal-dual solutions by applying variants of Newton's method to the first order optimality conditions and modifying the search direction and step lengths so that variables are strictly positive at each iteration [69].

Within a general framework, consider the quadratic problem

$$\begin{array}{ll} \text{minimize} & h'x + \frac{1}{2}x'Hx \\ \text{subject to} & Cx \leq v \\ & x \geq 0 \end{array} \tag{5.1}$$

Both the LTGP problem (2.24) and the relaxed problem (4.1) defined in the GP heuristic are covered by such a formulation. In this chapter matrix $C \in \mathbb{R}^{m \times n}$ merges the non-LMCs (as $-A$), the LMCs under consideration (B) and the variable upper bounds expressed as general inequality constraints, and v corresponds to their rhs. The number of variables n is $n_i \cdot n_u$, and the number of constraints m is the number of non-LMCs and the number of variables, $n_c + n_i \cdot n_u$, plus the number of LMCs considered for each problem solved.

The primal-dual pair of problem (5.1) where slack variables y and z are added to the primal and the dual, respectively, is:

$$\begin{array}{llll} \text{minimize} & h'x + \frac{1}{2}x'Hx & \text{maximize} & -v'g - \frac{1}{2}x'Hx \\ \text{subject to} & Cx + y = v & \text{subject to} & -C'g - Hx + z = h \\ & x,\, y \geq 0 & & g,\, z \geq 0 \end{array} \tag{5.2}$$

By introducing a logarithmic barrier term of parameter τ in the objective function in order to take into account the nonnegativity bounds, problems (5.2) can be solved through a barrier method starting from an *interior* point.

The first order optimality conditions of (5.2) with a logarithmic barrier of parameter τ for the nonnegativity bounds are then:

$$Cx + y = v \tag{5.3a}$$

$$-C'g - Hx + z = h \tag{5.3b}$$

$$Xz = \tau e \tag{5.3c}$$

$$Yg = \tau e \tag{5.3d}$$

$$(x, y, z, g) > 0 \tag{5.3e}$$

where the upper-case letters, X and Y, denote the diagonal matrices formed by the vectors x and y respectively.

Note that when τ is reduced to zero the first order optimality conditions are:

$$Cx + y = v \tag{5.4a}$$

$$-C'g - Hx + z = h \tag{5.4b}$$

$$Xz = 0 \tag{5.4c}$$

$$Yg = 0 \tag{5.4d}$$

$$(x, y, z, g) \geq 0 \tag{5.4e}$$

which correspond to the first order optimality conditions of (5.2).

Equations (5.4a) ensure *primal feasibility*, equations (5.4b) ensure *dual feasibility* and equations (5.4c – 5.4d) are the complementarity products.

5.1.1 The central path

The central path $\mathscr{C}$ is an arch of strictly feasible points that play a vital role in the theory of primal-dual algorithms. It is parameterized by the barrier parameter $\tau > 0$, and each point $(x_\tau, y_\tau, z_\tau, g_\tau) \in \mathscr{C}$ solves system (5.3). These conditions only differ from the first order conditions (5.4) with regard to the term τ on the right-hand side of (5.3c) and (5.3d). From (5.3), we can define the central path as

$$\mathscr{C} = \{(x_\tau, y_\tau, z_\tau, g_\tau) | \tau > 0\}.$$

The central path is unique and well defined if the interior of the primal and dual regions is non-empty [69].

5.1.2 Newton's method

Newton's method [65] is an iterative procedure for solving the problem of finding a solution to a set of nonlinear equations, $F(\zeta) = 0$. One step of the method is defined as follows. Given a point ζ, it forms a linear model for $F(\zeta)$ around the current point and obtains a search direction $\Delta\zeta$ by solving the following system of linear equations:

$$\nabla F(\zeta) \Delta\zeta = -F(\zeta)$$

where $\nabla F(\zeta)$ is the Jacobian of the set of nonlinear equations $F(\zeta)$. The new point ζ is then updated with $\Delta\zeta$.

Interior-point primal-dual methods use a barrier procedure that successively reduces the value of τ to find the solution to (5.2). A modification of Newton's algorithm is employed to solve the subproblem (5.3) for each τ:

- Only one iteration of Newton's algorithm is performed for each subproblem (for a given value of τ). If we let Newton's algorithm find the solution to (5.3) we will obtain a point in $\mathscr{C}$.
- Step lengths in the updating of the variables must sometimes be reduced in order to preserve the strict positivity of all variables (necessary in a barrier procedure).
- Some interior-point methods bias the search direction so they can move further in the direction before one of the components of the variables becomes negative.

5.1.3 Newton step direction

In order to describe the search direction, we introduce a *centering parameter* $\sigma \in [0, 1]$ and the *duality measure* μ defined by

$$\mu = \frac{x'z + y'g}{m + n}. \tag{5.5}$$

Parameter μ measures the average value of the pairwise products of the complementarity constraints. A certain fraction of μ is an adequate value for the logarithmic barrier parameter τ at a point given by x, y, z and g.

At each iteration of an infeasible primal-dual interior point method we solve one iteration of Newton's method applied to the nonlinear set of equations (5.3).

It can be easily deduced that the generic *Newton step* direction for problem (5.3a-5.3d) is:

$$\begin{bmatrix} -H & & I & -C' \\ C & I & & \\ Z & & X & \\ & G & & Y \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta g \end{bmatrix} = \begin{bmatrix} h + C'g + Hx - z \\ v - Cx - y \\ \sigma\mu e - Xz \\ \sigma\mu e - Yg \end{bmatrix}. \tag{5.6}$$

where $\tau = \sigma\mu$, with $\sigma = [0, 1]$, and X, Y, Z, G are diagonal matrices with vectors x, y, z, g, respectively.

If $\sigma = 1$, equation (5.6) defines a *centering direction*, i.e., a Newton step toward the point $(x_\mu, y_\mu, z_\mu, g_\mu) \in \mathscr{C}$, where all the pairwise products $x_i z_i$ and $y_i g_i$ are identical to μ. At the other extreme, the value $\sigma = 0$ gives the so-called *affine-scaling* direction, which attempts to reach the optimizer corresponding to $\tau = 0$. The value of σ is changed at each iteration of the algorithm.

5.1.4 Interior point steps

An infeasible interior-point method can begin from an arbitrary but necessarily interior point (all its components are strictly positive), which may be neither primal nor dual feasible, and it proceeds directly from there to the optimal solution. It simultaneously reduces primal infeasibility, dual infeasibility and complementarity.

We use vector ξ to denote the infeasibilities:

$$\xi_p = [v - Cx - y] \tag{5.7a}$$

$$\xi_d = \left[h + C'g + Hx - z\right] \tag{5.7b}$$

$$\xi_\mu = \begin{bmatrix} \sigma\mu e - Xz \\ \sigma\mu e - Yg \end{bmatrix} \tag{5.7c}$$

The optimality measure is the gap between the primal and the dual objective functions. It is easy to deduce that it corresponds to:

$$\text{gap} = x'\xi_d + g'\xi_p + (n + m)\mu \tag{5.8}$$

Infeasible primal-dual path-following method

i Define δ_c close to but strictly lower than 1, and initialize the variables (x, y, z, g) at a strictly interior point (> 0).

ii Compute the infeasibilities, ξ_p, ξ_d and ξ_μ, the barrier parameter μ and the gap.
If the gap is smaller than ϵ_g then END.

iii Determine σ and find the Newton direction Δ.

iv Compute a maximum step length for primal and dual variables: α_p, α_d.

v Update the variables:

$$x := x + \delta_c \alpha_p \Delta x \quad y := y + \delta_c \alpha_p \Delta y$$
$$z := z + \delta_c \alpha_d \Delta z \quad g := g + \delta_c \alpha_d \Delta g$$

and go to (ii)

5.1.5 Mehrotra predictor-corrector direction

The Mehrotra predictor-corrector technique [38] decomposes the step direction found in (5.6) into two parts:

$$\Delta = \Delta^{Mp} + \Delta^{Mc},$$

where Δ^{Mp} is the *predictor step* and Δ^{Mc} is the *corrector step*.

The predictor step Δ^{Mp} solves the affine-scaling direction, which points to the optimizer. Then, the centering parameter σ is chosen by assessing the quality of the Δ^{Mp} direction; and the error in the Δ^{Mp} direction is corrected, adding second-order information when solving the corrector step, Δ^{Mc}.

Therefore, to perform a single iteration of the predictor-corrector method we need two solutions from the same system for two different right-hand sides. Using direct factorizations of the system, the computation of more than one solution, in practice, adds little overhead to the algorithm but usually improves significantly the quality of the step direction.

Mehrotra predictor-corrector

i Solve the predictor step Δ^{Mp} (system (5.6) with $\sigma = 0$):

$$\begin{bmatrix} -H & & I & -C' \\ C & I & & \\ Z & & X & \\ & G & & Y \end{bmatrix} \begin{bmatrix} \Delta x^{Mp} \\ \Delta y^{Mp} \\ \Delta z^{Mp} \\ \Delta g^{Mp} \end{bmatrix} = \begin{bmatrix} \xi_d \\ \xi_p \\ -Xz \\ -Yg \end{bmatrix}. \tag{5.9}$$

ii Compute the maximum step lengths, α_p^{Mp} and α_d^{Mp}, and the μ^{Mp} parameter for the hypothetical point build with the direction Δ^{Mp}.

iii Set the centering parameter to $\sigma = \left(\frac{\mu^{Mp}}{\mu}\right)^3$.

iv Solve the corrector step Δ^{Mc}:

$$\begin{bmatrix} -H & & I & -C' \\ C & I & & \\ Z & & X & \\ & G & & Y \end{bmatrix} \begin{bmatrix} \Delta x^{Mc} \\ \Delta y^{Mc} \\ \Delta z^{Mc} \\ \Delta g^{Mc} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \sigma\mu e - \Delta X^{Mp}\Delta Z^{Mp} e \\ \sigma\mu e - \Delta Y^{Mp}\Delta G^{Mp} e \end{bmatrix} \tag{5.10}$$

where ΔX^{Mp} is the diagonal matrix of the vector Δx^{Mp}, and is equivalent for ΔZ^{Mp}, ΔY^{Mp} and ΔG^{Mp}.

v The search direction is $\Delta = \Delta^{Mp} + \Delta^{Mc}$.

Direction Δ is used in step (iii) of the *infeasible primal-dual path-following method* introduced in section 5.1.4.

5.1.6 Multiple centrality corrections

Mehrotra's predictor-corrector direction tries to reduce the μ parameter as much as possible and, assuming that a full step on the predictor direction will be done, it corrects the direction. Gondzio's multiple centrality correctors [28] try to modify a given direction in order to end at a point closer to the central path, which hopefully will result in a better direction in the next iteration.

Given a predictor direction Δ^P, such as the Mehrotra predictor-corrector direction, Gondzio's centrality corrector searches for a *centrality corrector direction* Δ^{Gc} such that larger stepsizes in primal and dual spaces are allowed for a composite direction

$$\Delta = \Delta^P + \Delta^{Gc}.$$

A centrality corrector direction does not aim at improving optimality but at homogenizing the pairwise complementarity products so that they fit in the interval $[\underline{\beta}\sigma\mu, \overline{\beta}\sigma\mu]$, where $\underline{\beta}$ and $\overline{\beta}$ are some relative threshold values for outlier complementarity products and σ is the centering parameter computed for the given predictor step.

Gondzio's multiple centrality correctors

i Assume that a predictor direction is given as Δ^P.

ii Compute the maximum primal and dual step length: α_p^P, α_d^P.

iii Suppose that the stepsizes will be enlarged to:

$$\begin{aligned} \widehat{\alpha}_p &:= \min(\alpha_p^P + \epsilon_h, 1) \\ \widehat{\alpha}_d &:= \min(\alpha_d^P + \epsilon_h, 1) \end{aligned} \tag{5.11}$$

where ϵ_h is a small amount (such as 0.1).

iv Compute the complementarity products for the trial point, $\widehat{x}$, $\widehat{y}$, $\widehat{z}$ and $\widehat{g}$, computed with the direction Δ^P and stepsizes $\widehat{\alpha}_p$ and $\widehat{\alpha}_d$.

v Define the *target* of the new complementarity pairwise products, ν_{xz} and ν_{yg}:

$$\nu_{x_k z_k} = \begin{cases} \overline{\beta}\mu & \text{if } \; x_k z_k > \overline{\beta}\mu \\ \underline{\beta}\mu & \text{if } \; x_k z_k < \underline{\beta}\mu \\ 0 & \text{otherwise} \end{cases} \qquad \forall\, k \tag{5.12}$$

and compute ν_{yg} equivalently.

vi Solve the system

$$\begin{bmatrix} -H & & I & -C' \\ C & I & & \\ Z & & X & \\ & G & & Y \end{bmatrix} \begin{bmatrix} \Delta x^{Gc} \\ \Delta y^{Gc} \\ \Delta z^{Gc} \\ \Delta g^{Gc} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \nu_{xz} \\ \nu_{yg} \end{bmatrix}. \tag{5.13}$$

The new direction is: $\Delta = \Delta^P + \Delta^{Gc}$.

The number of times that this procedure is repeated depends on the ratio of the costs of factorization and solution of the KKT system.

5.2 Solution of the Newton system

The most important and time-consuming part of an interior point method is computing the search direction. Any of the directions described solves a system such as:

$$\begin{bmatrix} -H & & I & -C' \\ C & I & & \\ Z & & X & \\ & G & & Y \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta g \end{bmatrix} = \begin{bmatrix} \theta_p \\ \theta_d \\ \theta_{xz} \\ \theta_{yg} \end{bmatrix} \tag{5.14}$$

After pivoting on the primal and dual slack variables, Δy and Δz, and rearranging the terms in order to obtain a symmetric indefinite system, the set of linear equations 5.14 reduces to the *augmented system*:

$$\begin{bmatrix} -(H + X^{-1}Z) & -C' \\ -C & G^{-1}Y \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta g \end{bmatrix} = \begin{bmatrix} \theta_p - X^{-1}\theta_{xz} \\ -\theta_d + G^{-1}\theta_{yg} \end{bmatrix} \tag{5.15a}$$

$$\text{with} \quad \Delta y = G^{-1}\theta_{yg} - G^{-1}Y\Delta g \tag{5.15b}$$

$$\text{and} \quad \Delta z = X^{-1}\theta_{xz} - X^{-1}Z\Delta x. \tag{5.15c}$$

System (5.15a) can be further reduced by pivoting either on the primal variables (Δx) or on the dual ones (Δg). The reduced form is known as the *normal equations*. The system to be factorized has a CC'- or $C'C$-like term in the matrix structure.

The choice of the form of the system depends on the data of the problem. The normal equations form gives a symmetric and positive definite coefficient matrix which can be factorized by Cholesky techniques. The coefficient matrix in (5.15a) is symmetric but indefinite, and the algorithms used for its factorization are more complicated. However, the augmented system has advantages in stability and flexibility [64, 65, 69].

5.2.1 The Newton system applied to the LTGP problem

When solving the LTGP problem with the GP heuristic we have to solve a succession of quadratic problems with an increasing number of inequality constraints. In this subsection we analyze the coefficient matrix C in order to justify our choice.

Matrix C is composed of the coefficients of the LMCs (C_L), the non-LMCs (C_{nL}) and the variable upper bounds (C_{ub}). We distinguish these parts in system (5.15a):

$$\begin{bmatrix} -(H + X^{-1}Z) & -C'_L & -C'_{nL} & -C'_{ub} \\ -C_L & G_L^{-1}Y_L & & \\ -C_{nL} & & G_{nL}^{-1}Y_{nL} & \\ -C_{ub} & & & G_{ub}^{-1}Y_{ub} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta g_L \\ \Delta g_{nL} \\ \Delta g_{ub} \end{bmatrix} = \begin{bmatrix} \widetilde{\theta}_p \\ \widetilde{\theta}_{d_L} \\ \widetilde{\theta}_{d_{nL}} \\ \widetilde{\theta}_{d_{ub}} \end{bmatrix} \tag{5.16}$$

Matrix C_L is sparse because each LMC belongs to an interval and its coefficients are 0 or 1. However, its size changes at each iteration of the heuristic (at each subproblem solved) and it can be quite large. Matrix C_{nL} has a fixed size n_c (for each case) but we do not have much information a priori about its structure, but it is likely that matrix C_{nl} will have multi-interval constraints. Matrix C_{ub} has fixed size $n_i \cdot n_u$ and it is the minus identity matrix. In view of the above considerations on the structure of the matrices we then pivot on variables Δg_L and Δ_{ub}:

$$\begin{bmatrix} -(H + X^{-1}Z + Y_{ub}^{-1}G_{ub} + C_L' Y_L^{-1} G_L C_L) & -C_{nL}' \\ -C_{nL} & G_{nL}^{-1} Y_{nL} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta g_{nL} \end{bmatrix} = \begin{bmatrix} \widetilde{\theta}_p - Y_{ub}^{-1} G \widetilde{\theta}_{d_{ub}} + C_L' Y_L^{-1} G_L \widetilde{\theta}_{d_L} \\ \widetilde{\theta}_{d_n L} \end{bmatrix} \tag{5.17a}$$

$$\text{with} \quad \Delta g_L = Y_L^{-1} G_L \widetilde{\theta}_{d_L} + Y_L^{-1} G_L C_L \Delta x \tag{5.17b}$$

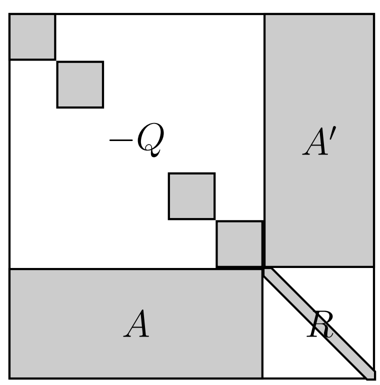
$$\text{and} \quad \Delta g_{ub} = Y_{ub}^{-1} G_{ub} \widetilde{\theta}_{d_{ub}} - Y_{ub}^{-1} G_{ub} C_{ub} \Delta x \tag{5.17c}$$

System (5.17a) has satisfactory properties:

- Matrix $(H + X^{-1}Z + Y_{ub}^{-1}G_{ub} + C_L' Y_L^{-1} G_L C_L)$ is block diagonal (with dense blocks because of the all-one LMCs).

- It has fixed dimension for all the subproblems produced by the heuristic.

- The sparsity pattern is



  where $Q \in \mathbb{R}^{n_i \cdot n_u \times n_i \cdot n_u}$ is positive definite, $A \in \mathbb{R}^{n_c \times n_i \cdot n_u}$ is the matrix of coefficients of the non-LMCs and $R \in \mathbb{R}^{n_c \times n_c}$ is diagonal and positive definite. Therefore, the matrix factorized is symmetric and quasidefinite, which as proved by Vanderbei [64], is a class of matrices strongly factorizable.

- If we pivot on variable Δg_{nL}, we cannot predict the sparsity pattern of the matrix, which may prove to be quite dense and large.

5.2.2 Solution of an indefinite system

There are two families of methods used for solving an indefinite system like (5.17a): an iterative one, such as the preconditioned conjugate gradient, or a direct method.

The conjugate gradient method solves a set of linear equations, $Px = b$, iteratively where the coefficient matrix is symmetric and positive definite. The preconditioned conjugate gradient uses another matrix M and leads to the problem $M^{-1}Px = M^{-1}b$ [59]. Matrix M is known as the *preconditioner* matrix, which approximates the coefficient matrix P and is easier to invert.

We use a direct method because we solve the same system several times with different right hand sides. Thus, the same factorization is reused. Also, an efficient preconditioner is not easy to find.

The Schur factorization

Matrix (5.17a) is factorized using the Schur complement expressed as:

$$\left[\begin{array}{c|c} -Q & A' \\ \hline A & R \end{array}\right] = \left[\begin{array}{c|c} L & \\ \hline -AL^{-'} & L_0 \end{array}\right] \left[\begin{array}{c|c} -I & \\ \hline & I \end{array}\right] \left[\begin{array}{c|c} L' & -L^{-1}A' \\ \hline & L_0' \end{array}\right] \tag{5.18}$$

where Q is factorized by Cholesky as $Q = LL'$ and $S = R + AQ^{-1}A'$ is the Schur complement, which is also positive definite and is factorized as $S = L_0 L_0'$.

The Schur complement, $S = R + A(LL')^{-1}A'$, is symmetric. However, we do not compute first $K = AL^{-1}$ and then multiply it, KK', but rather we perform the computation from right to left. This serves to obtain a more robust solution, as A is of moderate size in our case.

Given the structure of Q, which has n_i dense blocks of size n_u, the factor L consists of n_i Cholesky factorizations of matrices of size n_u.

Once we have the Schur factorization of the coefficient matrix of (5.17a), its solution is easy.

5.2.3 Control of system stability

When the interior point method is near the optimum, the Newton system can become ill-conditioned because the iterate may be too near of the border of the feasible area, with some variables given as near zero and others as very large. These values affect the diagonal of the solved matrix and create numerical difficulties when factorization is performed.

Several techniques have been applied to solve this problem. We have implemented the *regularization* method proposed by Altman and Gondzio [1]. Instead of factorizing system (5.17a) we factorize

$$\begin{bmatrix} -(H + X^{-1}Z + Y_{ub}^{-1}G_{ub} + C_L' Y_L^{-1} G_L C_L) & -C_{nL}' \\ -C_{nL} & G_{nL}^{-1} Y_{nL} \end{bmatrix} + \begin{bmatrix} -\Theta_p & \\ & \Theta_d \end{bmatrix} \tag{5.19}$$

where matrices Θ_p and Θ_d are positive-semidefinite and diagonal. Their components are selected at the moment when a pivot of the Cholesky factorization is found to be too small. Otherwise, these are zero matrices.

5.3 Computational results

Each solution to the problem using the GP heuristic requires the solution of many quadratic problems. These were solved using the infeasible primal-dual path-following method described in this chapter.

The interior point algorithm and the algebraic operations were coded in C. The main characteristics of the test cases solved are detailed in appendix A.1.

This section analyzes the performance of the interior point method implemented to solve the problems generated by the heuristic.

5.3.1 Solution of the test cases with the heuristic

Table 5.1 shows the same results as table 4.2 but the test cases were solved with the heuristic and the solver coded in C.

			Optimizer				Self-ord.	
	Obj. Func.	t (s)	Ite	act. LMCs	act. bo.	comp. LMCs	units u.b.	num. LMCs
ltp_01	9536489768.50	1.00	5	24	59	171	64	106
ltp_02	10961049207.89	0.70	8	48	22	117	25	22
ltp_03	10977720298.92	0.74	10	46	23	137	28	22
ltp_04	10979064734.23	0.81	11	51	24	147	29	22
ltp_05	8840160493.93	2.66	2	34	108	2293	119	1994
ltp_06	10667293289.59	4.03	2	176	82	2484	110	1994
ltp_07	5862769961.13	4.17	7	130	69	703	78	52
ltp_08	7077804310.06	2.14	7	85	76	411	97	48
ltp_09	5759806317.85	2.91	7	27	32	622	42	15
ltp_10	5267939457.49	0.77	8	39	9	213	17	13
ltp_11	4868047083.87	1.19	7	23	12	317	22	13
ltp_12	5122061425.65	3.49	1	243	74	794	74	15
ltp_13	6623277314.73	6.74	2	88	218	2386	237	52
ltp_14	4970958832.56	0.48	3	39	127	138	143	27
ltp_15	4740126251.94	0.57	5	40	40	169	66	27
ltp_16	6022480782.94	0.50	2	33	15	182	44	27
ltp_17	7050449790.96	0.75	10	40	37	241	76	27
ltp_18	4701104907.73	1.08	13	77	42	318	79	27
ltp_19	6999581735.01	1.76	10	80	112	354	170	27
avg		*1.92*	*6*	*70*	*62*	*642*	*80*	*238*

Table 5.1: Solution obtained with the GP heuristic

Results in table 4.2 were obtained using CPLEX whereas results in table 5.1 were obtained with the self-implemented interior-point solver. Due to the different optimality tolerance criteria of the solvers, some numbers change although the final objective function is practically the same.

5.3.2 Multiple-centrality corrections

This section compares performance when applying centrality corrections to the direction in some iterations (see section 5.1.6). At each iteration a maximum of four centrality corrections were applied.

To illustrate graphically the effect of the multiple centrality corrections in some directions we prepared a small instance. We identified some exact points determining the central path of a small problem for $C \in \mathbb{R}^{1\times 2}$. Figure 5.1 shows the feasible area (shaded part), with the level curves for a quadratic objective function. The red line is the central path for the primal variables, with an * mark at each point with μ equal to 10^4, 10^3, 10^2, 10, 1 and 0.1. We then solved the same problem with an interior point method using the Mehrotra direction and the Mehrotra direction with multiple centrality correctors. The starting point is primal and dual infeasible. Figure 5.1 shows the primal values at each iteration of the interior point solution of both solutions. In the first iteration the effect of the multiple centrality corrections that modify the Mehrotra direction – which is pointing towards an unfavorable area – can be observed. The example shown only corrects the Mehrotra direction in the first iteration.

The results in table 5.2 show the total time taken to solve each test case using the GP heuristic and the average number of interior point iterations performed during the solution of each subproblem which was finally solved, *ipm ite*. The number of problems solved for each case is shown in column *GP ite*.

The number of iterations is usually smaller when the Mehrotra predictor-corrector is recentered in some iterations. On average there is a saving of 3 iterations, though because both procedures require a similar computation time it is difficult to draw clear conclusions.

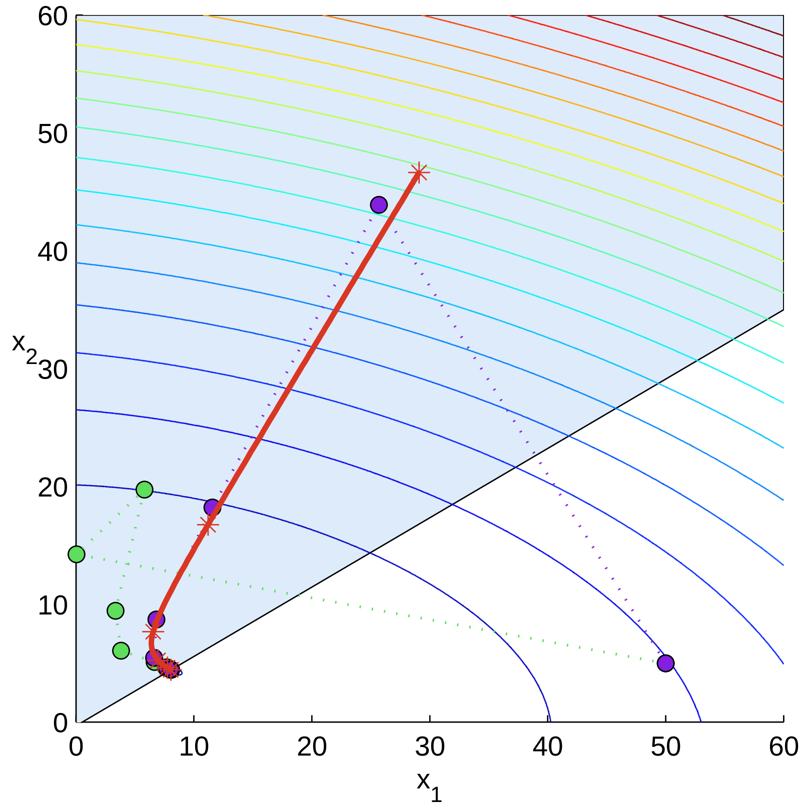


Figure 5.1: Central path $\mathscr{C}$ (red line with * markers) and interior point iterations using the Mehrotra predictor-corrector direction (dotted green line) and using Gondzio multiple centrality-correctors (dotted violet line).

	GP	Mehrotra		plus mcc		difference	
	ite	time	ite	time	ite	time	ite
ltp_01	5	0.69	18.4	1.00	15.2	-0.31	3.2
ltp_02	8	0.63	18.5	0.70	14.9	-0.07	3.6
ltp_03	10	0.78	19.2	0.74	13.9	0.04	5.3
ltp_04	11	0.90	19.7	0.81	14.2	0.09	5.5
ltp_05	2	2.66	19.5	2.66	15.5	0.00	4.0
ltp_06	2	4.04	22.5	4.03	18.0	0.01	4.5
ltp_07	7	4.52	16.4	4.17	13.7	0.35	2.7
ltp_08	7	2.48	20.5	2.14	15.9	0.34	4.6
ltp_09	7	2.38	17.2	2.91	17.1	-0.53	0.1
ltp_10	8	0.66	16.9	0.77	15.0	-0.10	1.9
ltp_11	7	0.94	16.8	1.19	15.3	-0.24	1.5
ltp_12	1	3.60	16.0	3.49	13.0	0.11	3.0
ltp_13	2	6.63	16.0	6.74	13.5	-0.11	2.5
ltp_14	3	0.49	16.3	0.48	14.7	0.00	1.6
ltp_15	5	0.70	16.8	0.57	14.2	0.14	2.6
ltp_16	2	0.56	13.5	0.50	12.0	0.06	1.5
ltp_17	10	0.72	17.0	0.75	14.2	-0.03	2.8
ltp_18	13	1.14	18.3	1.08	14.0	0.07	4.3
ltp_19	10	1.74	19.9	1.76	15.8	-0.02	4.1
average	*6.3*	*1.91*	*17.9*	*1.92*	*14.7*	*-0.01*	*3.1*

Table 5.2: Comparison of the total time (in seconds) and average number of iterations of the interior point method using multiple centrality correctors (*plus mcc*) or not.

Chapter 6

Warmstarting for interior point methods

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. We apply such technique to the sequence of problems generated by the GP heuristic.

The usual practice is to store an advanced iterate from the previous solution. Several strategies can then be followed in order to produce an initial point for the usual algorithm.

There are many papers on warmstarting for interior point methods, mainly for linear programming. Gondzio [29] applies warm-start techniques to a cutting plane scheme where new columns are appended. Yildirim and Wright [70] give two different approaches with the theoretical worst-case analysis, where the size of the problem does not change. Gondzio and Grothey [30] describe a new version of the warm start for linear programming and problems of constant size. This chapter 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 [5] 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 [6] 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 [21] develops a theoretical warm start for non-linear 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 initializing the new variables and for computing a direction to quickly recover primal and dual feasibility.

6.1 Warm-start Framework

Let us consider problem:

$$\begin{array}{ll} \text{minimize} & h'x + \frac{1}{2}x'Hx \\ \text{subject to} & Cx \leq v \\ & x \geq 0 \end{array} \tag{6.1}$$

which is one of the problems defined in chapter 5, as problem (5.1). From its solution and according to the GP heuristic, new constraints

$$\widetilde{C}x \leq \widetilde{v} \tag{6.2}$$

are generated. The new problem to be solved is then

$$\begin{array}{ll} \text{minimize} & h'x + \frac{1}{2}x'Hx \\ \text{subject to} & Cx \leq v \\ & \widetilde{C}x \leq \widetilde{v} \\ & x \geq 0 \end{array} \tag{6.3}$$

Our goal is to find the new optimizer of (6.3) as fast as possible using information collected from the previous solution to (6.1).

The equation system to be solved for the new problem at each iteration of an interior point method is:

$$Cx + y = v \tag{6.4a}$$

$$\widetilde{C}x + \widetilde{y} = \widetilde{v} \tag{6.4b}$$

$$-C'g - \widetilde{C}'\widetilde{g} - Hx + z = h \tag{6.4c}$$

$$Xz = \sigma\mu e \tag{6.4d}$$

$$Yg = \sigma\mu e \tag{6.4e}$$

$$\widetilde{Y}\widetilde{g} = \sigma\mu e \tag{6.4f}$$

where $\widetilde{C}$ is the matrix of coefficients for the new constraints and $\widetilde{v}$ its right hand side. New slack variables, $\widetilde{y}$, and the dual variables for the new inequality constraints, $\widetilde{g}$, are introduced. Again, $\widetilde{Y}$ is a diagonal matrix formed by vector $\widetilde{y}$.

The differences between the old problem (see equations (5.3)) and the new one (6.4) as regards the Newton system of the modified first order optimality conditions are:

- Regarding the variables, the afore-mentioned, new primal slacks, $\widetilde{y}$, and the corresponding dual variables, $\widetilde{g}$, of the new constraints.
- Regarding the constraints, the new primal constraints, $\widetilde{C}x + \widetilde{y} = \widetilde{v}$, a new term, $\widetilde{C}'\widetilde{g}$, in the dual constraints, and the complementarity condition for the new variables, $\widetilde{Y}\widetilde{g} = \sigma\mu e$.

6.2 Restoring Primal and Dual Feasibility

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 neighborhood of the central path. However, since this neighborhood is rather extensive, the iterates should simply stay away from the boundary.

As an illustration, we have re-solved the small problem presented in section 5.3.2 with one new constraint. Both central paths are plotted in figure 6.1. The extra constraint and the new central path are represented by the dotted lines.

Figure 6.1(a) shows the central paths for the primal variables (the slack variables y are implicit). An $*$ mark highlights the points with μ equal to 10^4, 10^3, 10^2, 10, 1 and 0.1. Figure 6.1(b) displays the dual variables in terms of μ.

In this example, the additional constraint does not cut across 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 point stored from the previous solution remains feasible, some special steps must always be taken.

Several procedures can be implemented to recover feasibility. 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 optimization, primal and dual feasibilities are related because of the term Hx, which appears in the dual feasibility constraint (6.4c).

Bearing in mind that the inclusion of a new constraint, even a feasible one, changes the central path, that is to say, the primal and dual variable values, our goal is to start the usual algorithm with a feasible and well-centered point.

6.3 Initial Point

Old variables

As initial values for the old variables (x, y, z and g) we choose the first iterate from the previous solution which has primal and dual relative infeasibility below a given threshold ϵ_c (e.g. $\epsilon_c = 0.001$) and also a small relative gap. However, the complementarity products 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 $[\gamma\mu, \mu/\gamma]$ (eg $\gamma = 0.1$). The stored point is known as the *μ-point*.

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 for an initial point in interior point methods is that the iterates have to be strictly positive. This gives a wide range of options for the initialization of the new variables, so we try to avoid degrading the first optimality conditions (6.4).

Because new primal constraints have been added (6.2), the magnitude of the primal infeasibility is beyond our control.

Given that an iterate that satisfies the first order optimality conditions for a given μ is chosen for the old variables, the dual infeasibility is $\widetilde{C}'\widetilde{g}$. If small positive values are chosen for the new dual variables, $\widetilde{g}$, dual feasibility is not significantly altered. In this case, 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 will have a large dual variable.

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

$$
\begin{aligned}
\widetilde{y}_j &= \begin{cases} \widetilde{v}_j - \widetilde{C}_j x & \text{if } \widetilde{v}_j - \widetilde{C}_j x > \epsilon_b \\ \max(\mu/\text{mean}(g), \epsilon_b) & \text{otherwise} \end{cases} \\
\widetilde{g}_j &= \mu/\widetilde{y}_j
\end{aligned} \tag{6.5}
$$

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 the old dual variables g. It has been observed that some components of g perform badly. They have large values compared with the rest. We excluded the outliers from the computation of the mean. These are the components such that: $g_j > \text{mean}(g) + 2 \cdot \text{stdev}(g)$, where $\text{mean}(g)$ is computed with all the components and $\text{stdev}(g)$ is the square root of the variance.

Parameter ϵ_b (e.g. $\epsilon_b = 100$) is a safeguard against small values of $\widetilde{y}$. This precaution is linked to the way the search direction is computed. The upper-left part of the factorized system (see section 5.2.1 for a detailed description) is:

$$
-(H + X^{-1}Z + Y_{ub}^{-1}G_{ub} + C_L' Y_L^{-1} G_L C_L + \widetilde{C}_L' \widetilde{Y}_L^{-1} \widetilde{G}_L \widetilde{C}_L). \tag{6.6}
$$

The choice of a small $\widetilde{y}_j$ value may worsen the condition of the initial matrix (6.6). Consequently, it is advisable to have the initial ratio $\widetilde{g}_j/\widetilde{y}_j$ under control. In view of the way we choose the $\widetilde{y}_j$ values the diagonal components $\widetilde{g}_j/\widetilde{y}_j$ are safely bounded in the first iteration: $\widetilde{g}_j/\widetilde{y}_j = \mu/\widetilde{y}_j^2 \leq \mu/\epsilon_b^2$. This is a satisfactory upper bound for an $\epsilon_b > 1$.

6.4 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 some iterations and concentrate on reducing primal and dual infeasibilities. The system requiring solution is

$$\begin{bmatrix} -H & & & I & -C' & -\widetilde{C}' \\ C & I & & & & \\ \widetilde{C} & & I & & & \\ Z & & & X & & \\ & G & & & Y & \\ & & \widetilde{G} & & & \widetilde{Y} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta \widetilde{y} \\ \Delta z \\ \Delta g \\ \Delta \widetilde{g} \end{bmatrix} = \begin{bmatrix} h + C'g + \widetilde{C}'\widetilde{g} + Hx - z \\ v - Cx - y \\ \widetilde{v} - \widetilde{C}x - \widetilde{y} \\ \mu e - Xz \\ \mu e - Yg \\ \mu e - \widetilde{Y}\widetilde{g} \end{bmatrix}. \tag{6.7}$$

This system is equivalent to the one that would be solved by the usual infeasible interior point method applied to (6.4) except that in this case no improvement to the complementarity condition is required ($\sigma = 1$).

6.4.1 Weighted Newton Step

It is unrealistic to assume that a full step in the direction obtained from (6.7) will be performed, for the reason that the variables must remain positive. It may occur that only a very small step is permitted, 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 (reducing the amount of infeasibility to be absorbed) and apply multiple centrality correctors [28] in order to provide a better chance for improvement in primal and/or dual feasibility.

Rather than solving (6.7) in one go, we split the right-hand side into three parts:

$$\begin{bmatrix} \xi_d \\ 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ \xi_p \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ 0 \\ \xi_\mu \end{bmatrix} \tag{6.8}$$

where the infeasibility in the new defined point is:

$$\widetilde{\xi}_d = \begin{bmatrix} h + C'g + \widetilde{C}'\widetilde{g} + Hx - z \end{bmatrix} \qquad \widetilde{\xi}_\mu = \begin{bmatrix} \mu e - Xz \\ \mu e - Yg \\ \mu e - \widetilde{Y}\widetilde{g} \end{bmatrix} \tag{6.9}$$
$$\widetilde{\xi}_p = \begin{bmatrix} v - Cx - y \\ \widetilde{v} - \widetilde{C}x - \widetilde{y} \end{bmatrix}$$

and solve the system for each part (using the same factorization). We obtain the directions Δ_d, Δ_p and Δ_μ. Note that Δ_p only attempts to recover primal feasibility: it leaves the dual infeasibility and the complementarity unchanged. Analogously, Δ_d recovers 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 infeasibility by this factor. The advantage of solving the system in three steps is that β can be chosen after solving the system.

Figure 6.2 shows the decomposition of the Newton step for the small case. The μ-point is an iterate from a previous solution. From this point the primal, dual and centering directions are computed and plotted for the primal variables x. In this case, the weighted Newton step is not needed.

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. 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. Primal infeasibility remains bounded by the primal step length, but the dual infeasibility reduction rate cannot be bounded. Given the initial point, the infeasible constraints are the new ones, giving a dual infeasibility $\xi_{\widetilde{d}}$. On moving along the direction computed from (6.7), the new dual infeasibility changes to $\widehat{\xi}_d$:

$$-C'(g + \alpha_d \Delta g) - \widetilde{C}'(\widetilde{g} + \alpha_d \Delta\widetilde{g}) - H(x + \alpha_p \Delta x) + (z + \alpha_d \Delta z) = h - \widehat{\xi}_d$$

and using the feasibility of the old point gives,

$$\alpha_d(-C'\Delta g - \widetilde{C}'\Delta\widetilde{g} + \Delta z) - \alpha_p H \Delta x = \widehat{\xi}_d,$$

where the dual step depends on the value of $H\Delta x$, which can be positive or negative.

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 centered ξ_μ will be nearly 0 in all its components, at least in the first iteration.

Final direction

When an estimate, albeit α_p^p or α_d^d, is very small, this is a sign that some components of this direction are not satisfactory. We hope that by using a weighted direction plus some 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:

$$\beta_p = \min(\kappa\alpha_p^p, 1) \tag{6.10a}$$

$$\beta_d = \min(\kappa\alpha_d^d, 1) \tag{6.10b}$$

$$\beta_\mu = 1 \text{ or } 0 \tag{6.10c}$$

with $\kappa > 1$. On the basis of previous computational experience we propose $\kappa = 2$, 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.

6.5 Outline of the warm-start procedure

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 (6.5). From this point the warm-start procedure is carried out and produces the initial point.

Warm-start procedure

i Compute the primal and dual infeasibilities, ξ_p and ξ_d, and the centrality deviations, ξ_μ

ii Compute the directions needed to recover primal and dual feasibility and centrality, solving system (6.7) with right-hand sides (6.9) $\rightarrow \Delta_p$, 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:

$$\begin{array}{lcl} \Delta_p & \rightarrow & \alpha_p^p, \alpha_p^d \\ \Delta_d & \rightarrow & \alpha_d^p, \alpha_d^d \end{array}$$

v Compute the direction weights:

$$\begin{array}{lcl} \beta_p & = & \min(\kappa \alpha_p^p, 1) \\ \beta_d & = & \min(\kappa \alpha_d^d, 1) \\ \beta_\mu & = & 1 \end{array}$$

vi Compute the predictor composite direction:

$$\Delta = \beta_p \Delta_p + \beta_d \Delta_d + \beta_\mu \Delta_\mu$$

vii Apply Gondzio multiple centrality correctors to Δ.

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

ix Compute primal and dual infeasibilities, ξ_p and ξ_d.

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

Otherwise switch to the usual interior point method.

6.6 Computational results

The warm-start approach presented was tested for the solution of the long-term generation planning using the GP heuristic. These problems are realistic test cases from the Spanish liberalized power pool. They are detailed in appendix A.1.

6.6.1 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 explained in chapter 5. The interior point 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 LTGP instances were:

- $\epsilon_c = 0.001$: if relative primal and dual infeasibility and the gap are smaller than ϵ_c then perform some centering steps and store the μ-point.

- $\kappa = 2$: scaling directions parameter.

- $\beta_\mu = 1$: 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.

- $\epsilon_w = 1e^{-4}$: if relative primal and dual infeasibility are less than ϵ_w the warm start procedure stops.

6.6.2 Heuristic statistics

The number of problems re-solved during the solution of the LTGP problem with the heuristic is reported in table 6.1. 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.

In each iteration of the heuristic a number of LMCs are added to the problem. We reoptimize when one or more of the LMCs generated are infeasible, but the new problem contains all previously generated constraints. In table 6.1, 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 6.2 details the relative infeasibility at each iteration of the heuristic where the new problem is re-solved. The 0 value indicates that the norm of the infeasibility is smaller than 10^{-6}. There is a significant difference between iteration 2 and the following ones (see chapter 4), because the iterative step-by-step stage of the heuristic starts at iteration 3.

6.6.3 Warm-start results

The results in table 6.3 show time (in seconds) and the number of iterations obtained by solving the LTGP instances with the heuristic. The way the interior-point solver begins is compared here: using a cold start or a warm start. The first problem solved is always initialized with a cold start (all the variable values are 10^6).

	num prb re-solved	LMCs 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 6.1: Number of problems re-solved starting from a warm-start point, and average number of new LMCs added and infeasible LMCs at the optimizer and at the stored μ-point.

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 done by the interior-point solver. The warm start results are detailed next: solution time, the average number of warm start iterations (*ws*), 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 is comparable, in terms of computational effort, to the number of iterations done with a cold start.

The last part of table 6.3 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). The average saving on interior point iterations is around 50%.

Seeing as the difference in time in most of the instances is a matter of fractions of a second (see table 6.3) and 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 imperceptible. We further analyzed how much of the time is employed by the routines that solve the problems (including the

	$\|\max(0, \widetilde{v}_j - \widetilde{C}x^*/\widetilde{v}_j)\|_2$										
	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		

Table 6.2: 2-norm of the relative infeasibility of the new constraints at each iteration where the problem is re-solved

warm start routine when applicable) using the `-pg` option of the gcc compiler and the `gprof` profiler [1]. Figure 6.3 shows the percentage of time employed by the solver function and the warm start routine (where applicable) and shows that 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 solution). This percentage is closely

[1] I 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 6.3: Time and average number of iterations done with a cold start or with a warm start

linked to the factorization of the Newton system routine. From the rest of the time, most of it is given over to the computation of the LMC right-hand sides.

Another conclusion that can be drawn form the analysis of the results is that the number of warm-start iterations not only depends on the magnitude of the infeasibilities, but also on the quality of the stored μ-point: the larger μ is, the better.

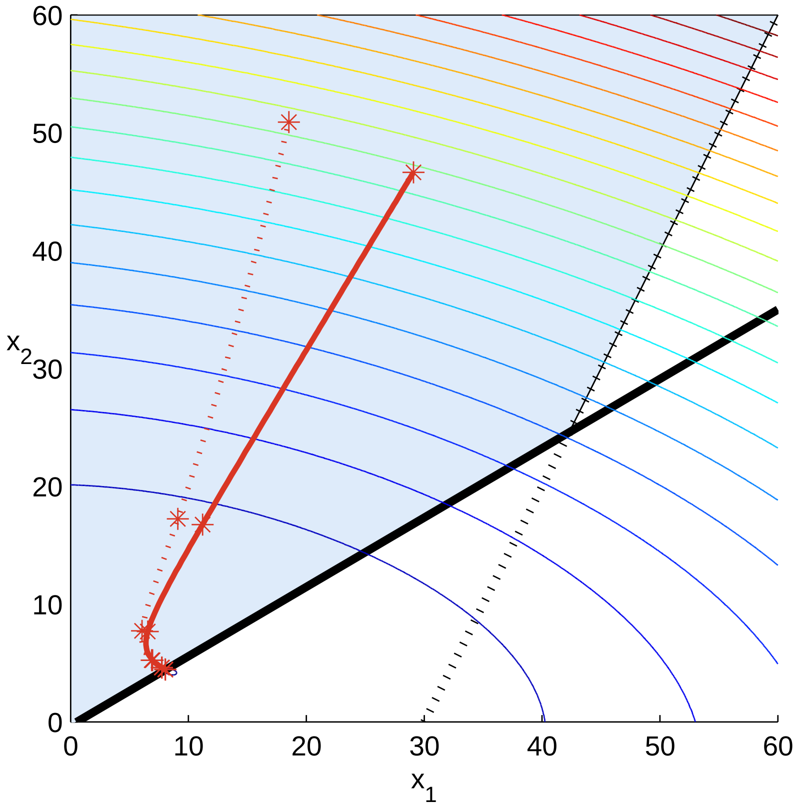
Importance of the μ-point

The analysis of the results presented in table 6.3 reveals that warmstarting in interior point methods is effective, although there is still room for improvement. In table 6.4 we show 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 taking care of the stored μ-point is also a key issue.

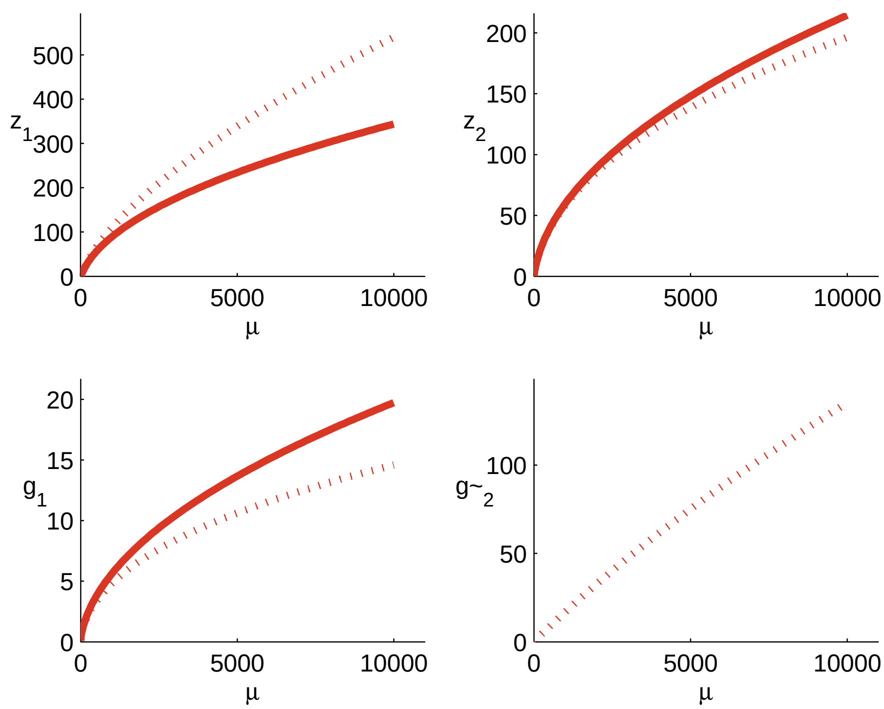
The small advantage presented by the warm-start procedure in the test cases solved may be a consequence of the shallowness of the appended cuts (LMCs). When problems arise with deeper cuts, the warm-start technique should present an advantage.

	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 6.4: Comparison of several initializing procedures

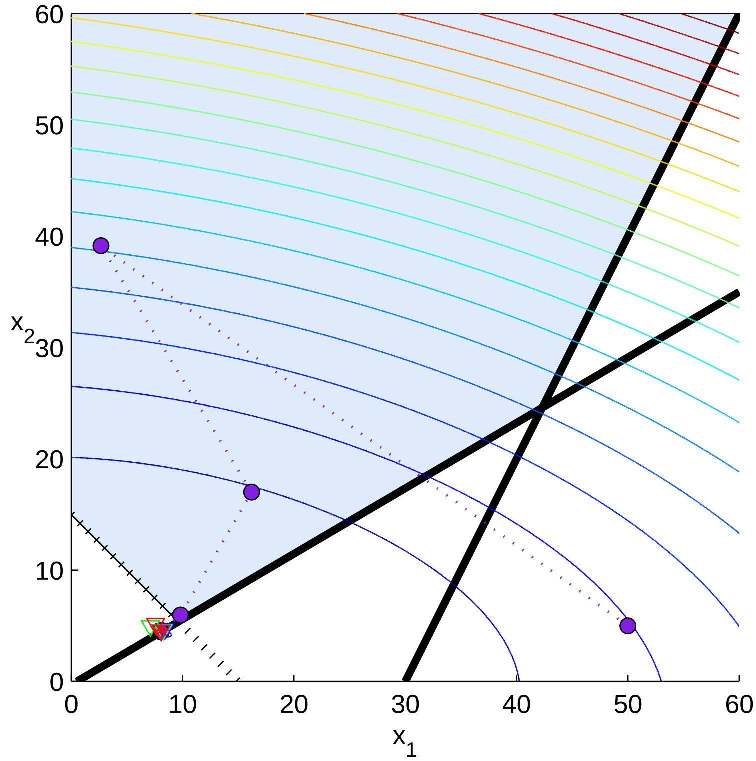


(a) Primal variables



(b) Dual variables

Figure 6.1: Changes in the central path when a new constraint is introduced (dotted line)



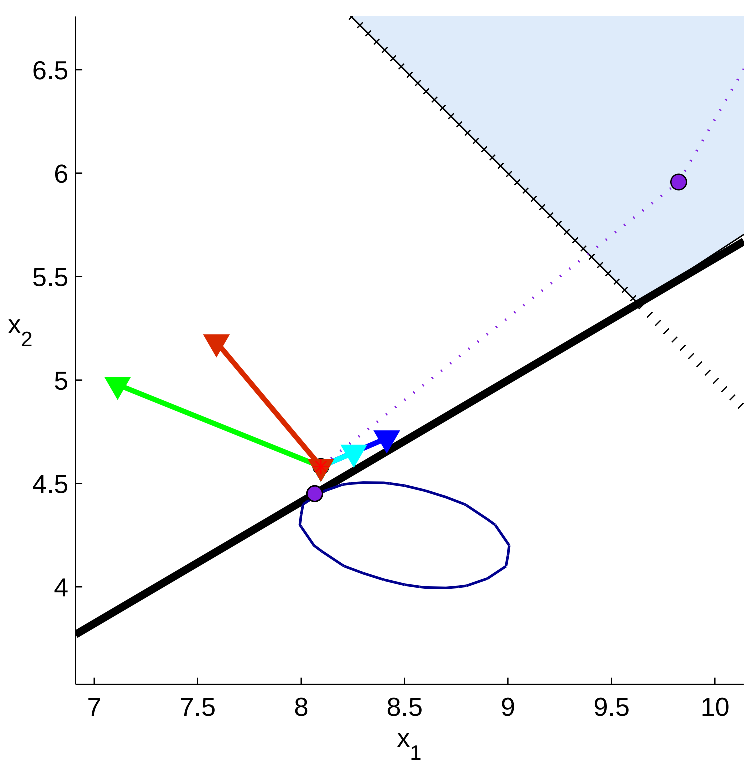


Figure 6.2: Interior point iterates, stored μ-point and zoom (below) of the step decomposition (blue: Δ_p, cyan: Δ_d, green: Δ_μ and red: $\Delta_p + \Delta_d + \Delta_\mu$) for recovering primal and dual infeasibility when a new constraint (dotted line) is considered.

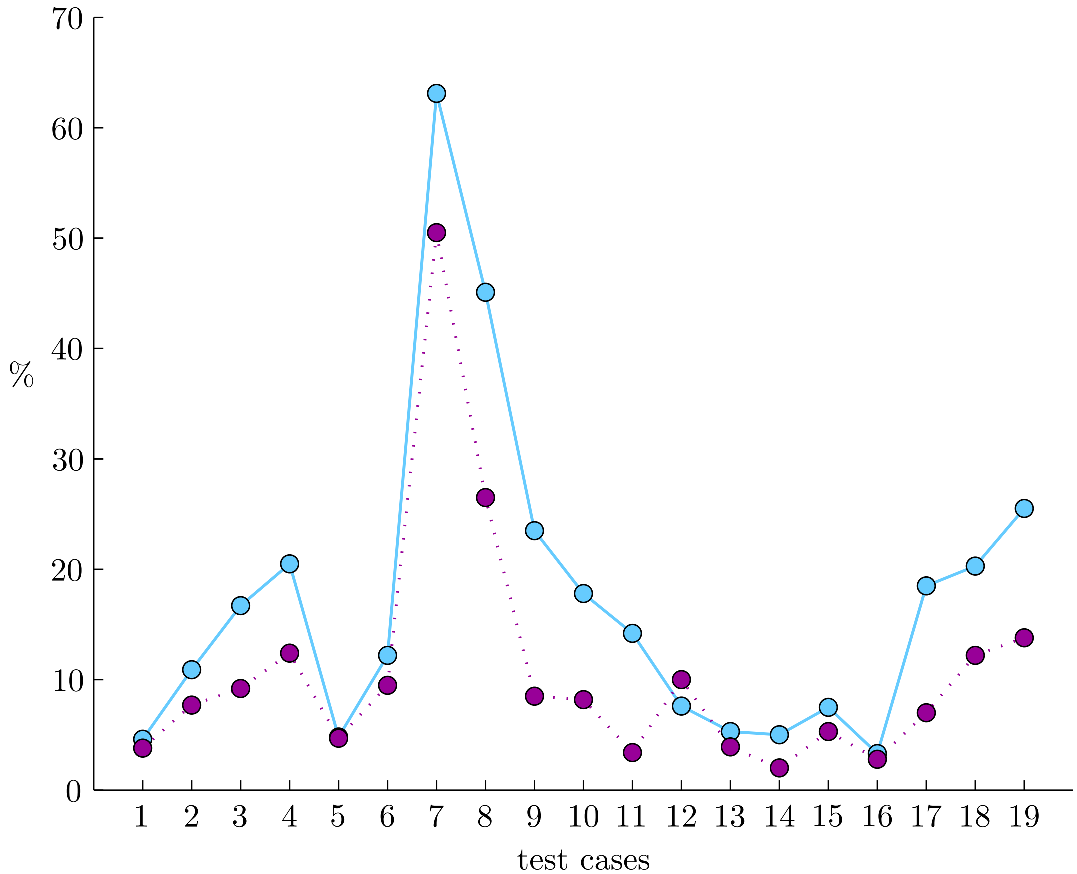


Figure 6.3: Percentage of time used by the interior-point solver in the GP heuristic (cold start represented by blue line and warm start by dotted violet line)

Chapter 7

Long- short- term coordination

Planning decisions are long- or short-term depending on their planning horizon. We consider long term to be a period of one or two years and short term up to one week. Should this be the current one or the following week, the parameters can all be identified or predicted with sufficient accuracy and, if it were not for market clearing prices, the short-term problem would be totally determined.

The detail included in representations of the power system and the time intervals diminishes as the planning horizon increases. Long-term decisions yield resource allocation requirements that must be incorporated into short-term scheduling. On the other hand, the short-term plan accurately models the operation of the unit generators (minimum and maximum power output, ramp-rate limits, ...) and the market rules, but it disregards the availability of resources. This coordination between the two decision levels is important in order to guarantee that certain aspects of the operation that arise in the long term are explicitly taken into account in the short term.

This problem has been addressed in several papers. In [23] there is a discussion concerning the use of *primal* and *dual* coordination between annual resource allocations and short-term operation. Reneses *et al.* [54] extend the range of possibilities with the *marginal resource-valuation functions* and compare these three approaches in a case study.

The *primal-information* approach imposes the production level of each resource projected by the long-term model on the short-term model in the form of constraints. The *dual-information* approach makes use of the dual prices of the constraints that limit the resources. The dual prices give marginal valuations for the resources. They are incorporated into the short-term objective function, penalizing or encouraging the use of the resource. The *marginal resource-valuation function* is a continuous valuation of a resource for a range of operating points that the company could face. The valuation function is an extension of the dual-information approach. The latter only provides one point of the function.

The main advantages of the primal approach are that it is easy to implement and that it ensures the feasibility of the long-term planning. However, it does not allow deviations from the forecasted parameters. Therefore, situations far removed from the predicted one may yield inconsistent or inefficient solutions. The dual approach is more flexible than the primal one, as the dual prices do not limit generation but rather lay out guidelines on the type of resources to be used. The main disadvantage of dual coordination is its lack of robustness. Small changes in the long-term parameters may lead to important changes in the dual prices.

We preferred a primal-information type approach with a tolerance. The units used in the short-term planning model also appear in the long-term method employed. The short-term planning period (usually one week) has been systematically included in the long-term planning model as the first long-term interval. Then the long-term results for its first interval are passed to the short-term model as constraints for the whole generation of each unit in the short-term period.

In mostly thermal power pools, where there is unlimited availability of all type of fuels, there is no much point in coordinating long- and short-term planning, and short-term planning then drives the long-term one, whose function is primarily to allow budgeting and scheduling fuel acquisitions. However, should there be an important share of hydro generation and long-term pumped storage plants, or should there be multi-interval limits in fuel supply or use, for example, take-or-pay fuel contracts, emission allowance constraints, power warranty constraints, or limited imports of certain fuels, as is the case in the Spanish power pool, the long- short-term coordination is a must, as the main opportunities for profits come from the long- and not from the short-term planning. It would then be nonsense overriding the long-term results with short-term operation.

The little attention paid to this subject in the literature on electricity markets may be due to the non-existence of a comprehensive long-term generation planning tool adapted to the electricity markets able to include all types of long-term constraints, and obtaining one such procedure was the main endeavor of this thesis.

The following sections present a short-term planning model for an SGC that operates in a pool-based market. It operates in coordination with the long-term planning model employed.

The LTGP problem must be systematically solved prior to solving a short-term planning problem in order to provide it with the relevant long-term reference frame, indicated by the long- and short-term coordination mechanism employed. Given that the long-term generation planning solution is quick to obtain, whenever the forecasted conditions change, the LTGP problem is re-solved and the short-term planning problem, with fresh LTGP information, is solved as well.

In view of the fact that this chapter develops a short-term model it uses some special notation. For example, parameter n_i still stands for the number of intervals, but it refers to the short-term ones. Another example is parameter t^i which stands for the length of the interval (in hours). In the short-term model t^i is always considered one hour. However, new notation has been used for parameters that may slightly change in meaning, as in the case of the number of units in the model, where n_u has not been used.

Further details on some parts of the model are given in [47].

7.1 The short-term planning model

When the generation unit capacity of an SGC is only a small percentage of the total capacity of all units participating in the pool, the behavior of this SGC cannot significantly influence the market price. It is then said that this SGC is a *price taker* (as opposed to a *price maker* whose behavior can influence the market price). The method of optimization of the production of a price taker in the short term is simpler than that of a price maker because, given a prediction of the hourly market prices, the production of each of the price taker's units can be optimized separately. The short-term procedure proposed in this thesis is appropriate for SGCs that are to a greater or lesser extent price makers.

The procedure presented in this chapter finds an estimate of the hourly market prices for a weekly period through the solution of a double optimization process: a unit commitment in which the zero-priced bids of the units are determined, and a nonlinear continuous schedule in which the generations are refined. It employs two types of prediction of the supply-bid function for every hour: the first linear and static, and the second nonlinear, with the linear term being a linear function of unit generations.

In [11] a two-stage procedure is proposed for optimizing the profit of a price taker. For each of its units, a self-scheduling problem using a linear profit function with an estimate of the expected hourly prices is first solved through integer programming. In the second stage, and using the former results, an optimal bidding policy for each hour is established taking into account the probability distribution of market prices, assumed to be lognormal. An extension of this procedure, which accounts for the loss-of-profit risk, can be found in [12].

The model takes into account all participants in the generation bidding process, and considers as data the long-term results obtained previously for the week to be planned, since the short-term planning is hierarchically coordinated with long-term planning results; therefore, constraints in the short-term optimization were included to enforce this.

The emphasis here is placed on the feasibility and optimality of the profit maximization sought after by all the participants in the liberalized pool. It is therefore assumed that an SGC, besides its own generation portfolio, has sufficient knowledge about the generation units of the rest of the market agents. Part of this information is publicly available in the bulletins released by the System Operator. The rest of it can be approximately inferred from other public sources or by comparison with similar units. (This is the case in the Spanish power pool.)

7.1.1 Generation units in short-term planning

Both hydro and thermal generation are modeled more accurately by short-term planning than by long-term planning. Concerning the SGC the units considered are:

- each thermal unit that participates in the auction process during the weekly period, and
- a hydro unit for each reservoir system with full details of storage and discharge.

In the case of the rest of generating companies:

- units from the same company with similar characteristics (e.g. the fuel used) are merged as a single pseudo-unit, and
- the hydro-systems of the competitor generation companies are considered as one or more pseudo-thermal units.

In conclusion, there will be n_t thermal (or pseudo-thermal) units and n_h hydro units. The units considered in the short-term planning model are also included in the long-term one. In the long-term model there may be some more units (those units that are in maintenance in the short-term period), and the SGC hydro units are grouped into one pseudo-thermal unit.

Hourly wind-power generation during the short-term weekly period should be predicted and deducted from the forecasted load.

Thermal units and pseudo-units

The relevant parameters of thermal unit j are:

- $\overline{c}_j$ and $\underline{c}_j$: maximum and minimum power capacity
- f_j: linear generation cost

and should the unit be susceptible to start-up and shut-down:

- $n_{\mathrm{on}\,j}$ and $n_{\mathrm{off}\,j}$: minimum operation time and minimum idle time
- $f_{\mathrm{su}\,j}$: start-up cost

Variables and parameters corresponding to a certain hourly interval in the short-term period will be denoted by a supra-index ^{i} indicating the interval number. Let n_i be the total number of intervals considered. We will normally consider a weekly period subdivided into $n_i = 168$ hourly intervals, but shorter periods could also be considered.

Let g_j^i be the power generation of thermal unit j over interval i, and $g^i = \sum_{j=1}^{n_t} g_j^i$ the total thermal generation in interval i.

The hydro generation

The average generated hydropower h_k^i over interval i of duration t^i (one hour) in reservoir k will be:

$$h_k^i = \frac{\rho_k \mathrm{g}}{t^i} y_k^i \widetilde{\sigma}_k^i$$

where $\rho_k < 1$ is the efficiency of the turbine-alternator system, g is the acceleration due to gravity, y_k^i is the volume of water discharged over interval i, and $\widetilde{\sigma}_k^i$ is the equivalent water head:

$$\widetilde{\sigma}_k^i = \sigma_{\mathrm{b}k} + \frac{\sigma_{\mathrm{l}k}}{2}(v_k^{i-1} + v_k^i) + \frac{\sigma_{\mathrm{q}k}}{3}(v_k^i - v_k^{i-1})^2 + \sigma_{\mathrm{q}k} v_k^{i-1} v_k^i + \frac{\sigma_{\mathrm{c}k}}{4}({v_k^{i-1}}^2 + {v_k^i}^2)(v_k^{i-1} + v_k^i) \qquad (7.1)$$

where v_k^i is the volume of water in reservoir k at the end of interval i. Parameters $\sigma_{\mathrm{b}k}$, $\sigma_{\mathrm{l}k}$, $\sigma_{\mathrm{q}k}$, and $\sigma_{\mathrm{c}k}$ are respectively the basic, linear, quadratic and cubic coefficients of the head to volume function, which are data for the problem.

The water balance in reservoir k over interval i in a cascaded reservoir basin would be:

$$v_k^{i-1} + \mathrm{w}_k^i + \sum_{j \in \mathcal{H}_k} (y_j^i + p_j^i) = v_k^i + y_k^i + p_k^i \tag{7.2}$$

where w_k^i is the natural inflow, $\mathcal{H}_k$ the set of reservoirs directly upstream of reservoir k, and p_k^i the spillage. Inflows w_k^i are forecasted.

The total hydro generation in interval i is

$$\begin{aligned} h^i = \sum_{k=1}^{n_h} h_k^i = \quad & \frac{\mathrm{g}}{t^i} \sum_{k=1}^{n_h} \rho_k \Big\{ \sigma_{\mathrm{b}k} + \tfrac{\sigma_{\mathrm{l}k}}{2}(v_k^{i-1} + v_k^i) + \tfrac{\sigma_{\mathrm{q}k}}{3}(v_k^i - v_k^{i-1})^2 + \\ & \sigma_{\mathrm{q}k} v_k^{i-1} v_k^i + \tfrac{\sigma_{\mathrm{c}k}}{4}({v_k^{i-1}}^2 + {v_k^i}^2)(v_k^{i-1} + v_k^i) \Big\} y_k^i \end{aligned} , \tag{7.3}$$

which is nonlinear.

The evolution of water volume in reservoirs, over n_i successive intervals, can be modeled through a replicated hydro network [56]. The initial volume in reservoir k during interval i is the final volume of this reservoir in interval $i-1$. Discharges y_k^i and spillages p_k^i are also flows on arcs of the replicated network, and its node balances are expressed in equations such as (7.2). Initial and final water volumes in reservoirs in the first and last intervals, v_k^0 and $\mathrm{v}_k^{n_i}$ respectively, will be considered data for the problem. (Initial volumes are usually the current ones, and final volumes are the target values obtainable from long-term generation planning.)

Linearization of hydro-generation

Equation (7.3) is a fourth-order polynomial of hydro variables, and may make the generation optimization problem hard to solve. One way to simplify the problem is to linearize these equations as follows: the term in braces in (7.3), which is the $k^{\underline{\mathrm{th}}}$ reservoir head over interval i, will be considered to be a constant $\widehat{\sigma}_k^i$, as if the volumes $\widehat{v}_k^i$ involved in its calculation were known quantities,

$$\widehat{\sigma}_k^i = \sigma_{\mathrm{b}k} + \frac{\sigma_{\mathrm{l}k}}{2}(\widehat{v}_k^{i-1} + \widehat{v}_k^i) + \frac{\sigma_{\mathrm{q}k}}{3}(\widehat{v}_k^i - \widehat{v}_k^{i-1})^2 + \sigma_{\mathrm{q}k}\widehat{v}_k^{i-1}\widehat{v}_k^i + \frac{\sigma_{\mathrm{c}k}}{4}((\widehat{v}_k^{i-1})^2 + {\widehat{v}_k^{i}}{}^2)(\widehat{v}_k^{i-1} + \widehat{v}_k^i)$$

and the succession of volumes employed, $\widehat{v}_k^i$, $i = 1 : n_i$, could be a past solution, or a uniform variation from the initial v_k^0 to the final volume $v_k^{n_i}$. Thus, equations (7.3) become:

$$\begin{aligned} h_k^i = \quad & \frac{\rho_k \mathrm{g} \widehat{\sigma}_k^i}{t^i} y_k^i & \forall\, i,\ \forall\, k \\ h^i = \quad & \sum_{k=1}^{n_h} h_k^i = \frac{\mathrm{g}}{t^i} \sum_{k=1}^{n_h} \rho_k \widehat{\sigma}_k^i y_k^i & \forall\, i \end{aligned} \tag{7.4}$$

Those equations (7.4), which are linear in the discharges y_k^i, together with the balance equations (7.2) constitute the linearized hydro-generation model.

7.1.2 Coordination with the long-term planning results

Long-term and short-term must be coordinated to ensure that certain aspects of the operation that arise in the long-term model are explicitly taken into account in the short-term, so that in no case will the short-term scheduling override the long-term results.

Information is exchanged by means of a primal-information approach with a tolerance. The units and pseudo-units used in the short-term planning already appear in the long-term method employed. The short-term planning period (usually one week) was systematically included in the long-term planning as the first interval. Then the results for the first long-term interval were passed to the short-term as constraints for the whole generation of each unit in the short-term period.

Let x_j^1 be the expected energy generated by unit j over the first long-term period, obtained by the long-term planning procedure. It is then necessary to impose

$$(1-\delta_{l2s})x_j^1 \leq \sum_{i=1}^{n_i} t^i g_j^i \leq (1+\delta_{l2s})x_j^1 \qquad j=1,\ldots,n_t \tag{7.5}$$

in the short term, where δ_{l2s} is a small positive tolerance that must be employed because, in long-term studies, energies obtained take into account outage probabilities, which are not considered in the short term.

Regarding hydro generation, although a model was employed where a whole hydro generation basin is considered as a thermal unit plus a total generation constraint, a more detailed long-term hydro generation model which accounts for each individual reservoir is also possible [31]. The stored volumes of water in each reservoir at the end of the first long-term interval are passed on as final values to the short-term problem. The initial stored volumes (current) in each reservoir are the same for long- and short-term planning. If a single reservoir or a hydro generation basin was modeled as a pseudo thermal unit plus a total generation constraint, a generation constraint for the reservoir or the basin would have to be introduced into the short-term model to ensure that hydro generation occurs in the first long-term interval.

7.1.3 The supply-bid function

Generation bids corresponding to a certain past hour ordered by increasing price have a characteristic shape. The function giving the generation price (in €/MWh) for each energy unit bid will be referred to as the *supply-bid function*. The most important parameters are the following:

- Generation companies bid a considerable part of the capacity of many of their generators at zero price.
- The sum of the hourly zero-priced generation bids of all companies falls below the hourly load but amounts to a large part of it.
- The shape of the supply-bid function in the zone corresponding to positive prices is irregular. It can be reasonably approximated by a polynomial of degree four or higher.

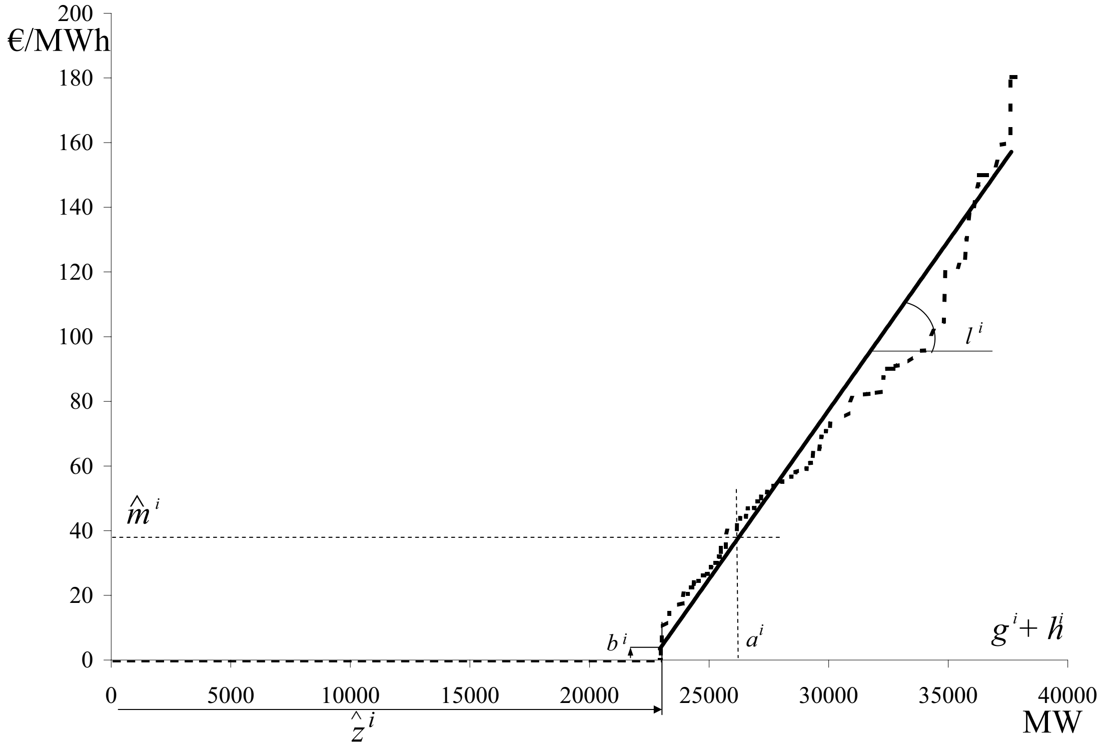


Figure 7.1: Supply-bid function corresponding to 10 to 11 a.m. on a Monday in March 2004 in the Spanish power pool (dashed line) and linearized supply-bid function (continuous line).

The intersection of the supply-bid function of a given hour with the demand function (in terms of price) or with the forecasted demand for this hour gives the forecasted *market price* m^i. Figure 7.1 shows the supply-bid function for the interval between 10 and 11 a.m. on a Monday in March 2004 in the Spanish pool.

Although there is a demand-bid function for each interval, its prediction is as complex as the prediction of the matched load a^i in each interval. The predicted matched load a^i was preferred because it makes the model simpler.

An important variable, necessary for the determination of the supply-bid function, is the amount of zero-priced energy z^i in the interval. One of our goals is to obtain a good estimate of z^i for each hourly interval.

Generation bids and zero-priced energies can be expressed in MW, as is the load a^i, because the duration of all intervals considered is one hour.

The zero-priced supply bids

Our experience with the Spanish power pool shows that the proportion of zero-priced energy bids $\widehat{z}_i$ is generally over 65% of the load to be supplied a^i. Furthermore, the amount of zero-priced bids has similar patterns for several units and hours:

- Committed thermal units contribute to $\widehat{z}^i$ with $\chi^i_j * \overline{c}_j$, where χ^i_j is the proportion of zero-priced energy offered with respect to maximum capacity.

- For hydro units the amount of zero-priced bids is proportional to the forecasted demand: $z^i_k = \lambda_k a^i$.

The coefficients of the contribution to the zero-priced supply curve can be calculated from past records, which may vary in some intervals (different patterns for the peak and the base hours can be easily distinguished). Further details on the modeling, with a graph-based rationalization, can be found in [47].

Linear approximation to the supply-bid function

Figure 7.1 shows the linearized supply-bid function in interval i. It is determined by the estimates $\widehat{z}^{\,i}$ of the zero-priced energy in the interval, and by the basic and linear coefficients, b^i and l^i, giving the approximate linearized supply-bid function for the non-zero-priced part. The linearized supply-bid function is then:

$$m^i = b^i + l^i(g^i + h^i - \widehat{z}^{\,i}) \qquad \text{for} \;\; g^i + h^i \geq \widehat{z}^{\,i}$$

and $m^i = 0$ otherwise. Given that the market always clears, the value m^i of the linearized supply-bid function for the load $a^i = g^i + h^i$ is the market-price estimate:

$$\widehat{m}^i = b^i + l^i(a^i - \widehat{z}^{\,i}) \qquad \text{for} \;\; a^i \geq \widehat{z}^{\,i} \tag{7.6}$$

and $\widehat{m}^i = 0$ otherwise, being $\widehat{z}^i = \sum_{j=1}^{n_t} \widehat{z}^i_j + \sum_{k=1}^{n_h} \widehat{z}^i_k$ the total zero-priced supply bids.

This market-price estimate derived from the linearized supply-bid function will be employed in the solution of a unit commitment problem.

Nonlinear approximation to the supply-bid function

The non-zero part of the supply-bid function adjusted by a fourth-degree polynomial instead of a regression line is expressed as

$$\begin{aligned} m^i = \; & \widetilde{l^i}(g^i + h^i - \widehat{z}^{\,i}) + \gamma^i_q(g^i + h^i - \widehat{z}^{\,i})^2 + \\ & \gamma^i_c(g^i + h^i - \widehat{z}^{\,i})^3 + \gamma^i_t(g^i + h^i - \widehat{z}^{\,i})^4 \qquad \text{for} \;\; g^i + h^i \geq \widehat{z}^{\,i} \end{aligned}$$

and $m^i = 0$ otherwise, with $\widetilde{l^i}$ defined as $\beta^i_0 + \sum_{j=1}^{n_t} \beta^i_j(g^i_j - \widehat{z}^i_j)$. Then, the market-price estimate using the nonlinear supply-bid function, for $a^i = g^i + h^i$, is

$$\begin{aligned} \widetilde{m}^i = & \ \widetilde{l}^i(a^i - \widehat{z}^{\,i}) + \gamma_q^i(a^i - \widehat{z}^{\,i})^2 + \gamma_c^i(a^i - \widehat{z}^{\,i})^3 + \gamma_t^i(a^i - \widehat{z}^{\,i})^4 \\ = & \ \widetilde{l}^i(a^i - \widehat{z}^{\,i}) + \Gamma^i \\ = & \ \{\beta_0^i + \textstyle\sum_{j=1}^{n_t} \beta_j^i(g_j^i - \widehat{z}_j^i)\}(a^i - \widehat{z}^{\,i}) + \Gamma^i \end{aligned} \quad (7.7)$$

where the thermal and hydro generation, g_j^i and h_k^i, will be determined by optimization of the model. This function, linear in g_j^i and h_k^i, is used in a refinement stage, where $\widehat{z}^{\,i}$ and $\widehat{z}_j^i$ are data.

The basic coefficient of the curve fitting, $\widetilde{l}^i$, rather than being taken as fixed parameter (as are the rest of coefficients γ_q^i, γ_c^i and γ_t^i), may change. Coefficient $\widetilde{l}^i$ has a constant part β_0^i, and a variable part β_j^i that changes with the thermal generation g_j^i through the linear expression (7.7); hydro generation is not considered to influence the slope $\widetilde{l}^i$. The computation of the β coefficients is explained in [47].

Although the long- to short-term coordination constraints (7.5) limit the overall generation of each unit over the short-term period, they leave enough scheduling freedom within the short-term hours for finding the best way to maximize profits by influencing the market prices through the supply bid function presented (7.7), therefore, the strategic oligopolistic equilibrium can be reached.

Figure 7.2 shows the polynomial fit and how it changes when, through changes in thermal generation g_j^i, $\widetilde{l}^i$ is increased or decreased by 10%.

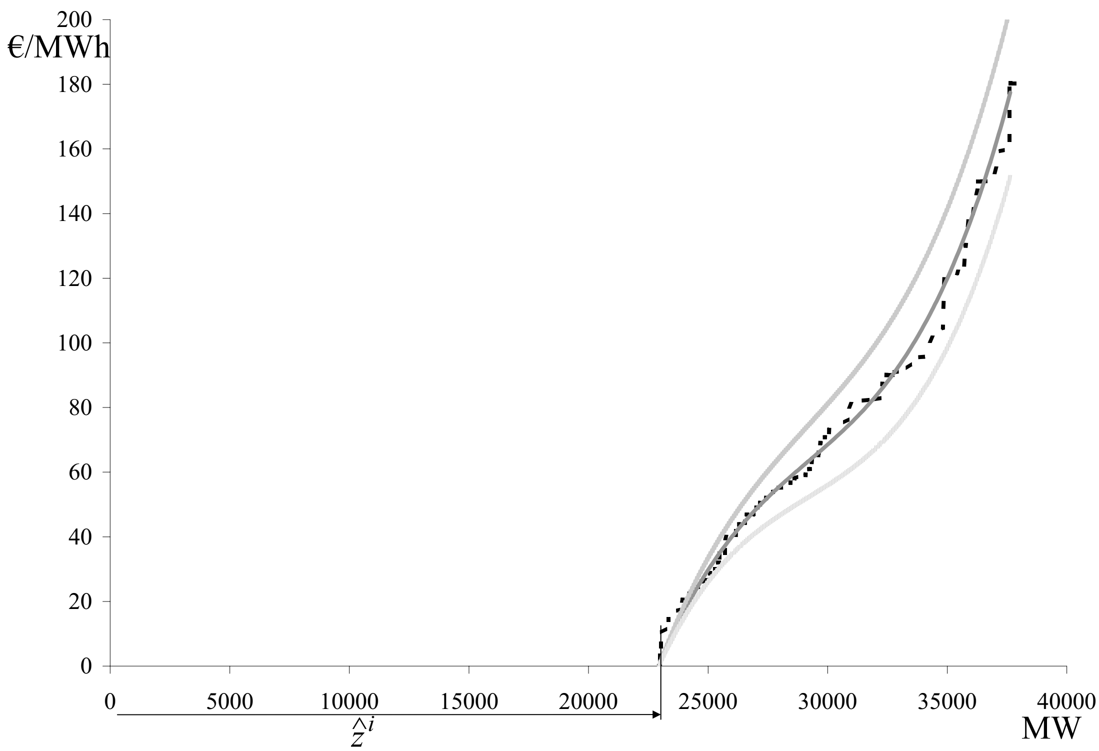


Figure 7.2: Nonlinear supply-bid function for interval i. (The original supply-bid function is the dashed curve.)

Prevention of collusion

Given that the slope in the supply-bid function, $\widetilde{l^i}$, has a direct influence on the overall profit (this being maximized), it is necessary, in order to prevent the effect of collusion between participants, to place a constraint on the changes in the slopes with respect to the original ones $\widetilde{l^i}$, which are a forecast deduced from available data. The type of constraint introduced is

$$-\epsilon_\beta \leq \sum_{i=1}^{n_i} \left\{ \widetilde{l^i} - \beta_0^i - \sum_{j=1}^{n_t} \beta_j^i (g_j^i - \widehat{z}_j^i) \right\} \leq \epsilon_\beta \tag{7.8}$$

where ϵ_β is a small tolerance.

7.1.4 Formulation of the unit commitment

The following formulation, that follows on from the one reported in [48], expresses that the commitment of the generation units (those thermal units belonging to the SGC) will respect the start-up and shut-down periods.

The notation used here resembles that used for the formulation of the maintenance schedule.

Let Φ be the subset of units to be committed. Let $s_j^i \in \mathbb{B}$ ($\mathbb{B} = \{0, 1\}$) be a binary variable expressing the off-on operating status of the thermal unit j over the interval i. The generation g_j^i of the thermal unit j in interval i will satisfy

$$\underline{c}_j \, s_j^i \leq g_j^i \leq \bar{c}_j \, s_j^i \qquad \forall\, i \quad \forall\, j \tag{7.9}$$

Values of s_j^i and s_j^{i+1} must obey certain operating rules that allow for the constraints of the minimum service time of $n_{\mathrm{on}j}$ hours and minimum idle time of $n_{\mathrm{off}j}$ hours. It is necessary to introduce two extra binary variables $d_j^i \in \mathbb{B}$ and $u_j^i \in \mathbb{B}$ for each s_j^i.

Variable u_j^i is a start-up indicator for the thermal unit j. It is zero for all the intervals except when unit j changes from $s_j^{i-1} = 0$ to $s_j^i = 1$. Similarly, d_j^i is a shut-down indicator for the thermal unit j. It is zero for all the intervals except when the unit j changes from $s_j^{i-1} = 1$ to $s_j^i = 0$. With u_j^i and d_j^i it is then easy to model the minimum up and down times of unit j. The following set of constraints:

$$s_j^i - s_j^{i-1} - u_j^i + d_j^i = 0 \qquad \forall\, i \quad \forall\, j \in \Phi \tag{7.10a}$$

$$u_j^i + \sum_{k=i}^{\min(i+n_{\mathrm{on}j}, n_i)} d_j^k \leq 1 \qquad \forall\, i \quad \forall\, j \in \Phi \tag{7.10b}$$

$$d_j^i + \sum_{k=i+1}^{\min(i+n_{\mathrm{off}j}, n_i)} u_j^k \leq 1 \qquad \forall\, i \quad \forall\, j \in \Phi \tag{7.10c}$$

uniquely define the binary variables, and ensure that the service and the idle times are at least $n_{\mathrm{on}j}$ and $n_{\mathrm{off}j}$.

7.2 Short-term models

This section gathers together the formulation presented in two models, one for each optimization stage.

7.2.1 Linearized unit commitment model

In order to obtain a unit commitment that maximizes profit, market price minus generation costs should be maximized

$$\text{maximize} \sum_{i=1}^{n_i} \left\{ a^i \, \widehat{m}^{\,i} - \sum_{j=1}^{n_t} (f_j \, g^i_j + f_{\text{su}j} \, s^i_j) \right\} \tag{7.11a}$$

$$\text{subject to } \widehat{m}^{\,i} = b^i + l^i \, (a^i - \widehat{z}^{\,i}) \quad \forall \, i \tag{7.11b}$$

where, taking the market price equation (7.11b) and the zero-priced bid policy presented in section 7.1.3, the objective function (7.11a) is linear in the binary variables s^i_j, d^i_j and u^i_j.

The reason we maximize the overall profit of all SGCs participating in the pool, and not just that of the SGC dealt with in detail, is because we assume that all participating SGCs are trying to maximize their profit at the same time.

The complete linearized unit-commitment model can be recast as:

$$\max \sum_{i=1}^{n_i} \{ a^i \, \widehat{m}^{\,i} - \sum_{j=1}^{n_t} (f_j \, g^i_j + f_{\mathrm{su}j} \, u^i_j) \} \tag{7.12a}$$

$$\text{s. t. } \widehat{m}^{\,i} = b^i + l^i \, (a^i - \widehat{z}^{\,i}) \qquad \forall i \tag{7.12b}$$

$$\widehat{z}^{\,i}_k = a^i \, \lambda^i_k \qquad \forall i \quad k = 1 : n_h \tag{7.12c}$$

$$\widehat{z}^{\,i}_j = \overline{c}_j \, \chi^i_j \qquad \forall i \quad j = 1 : n_t, \ j \notin \Phi \tag{7.12d}$$

$$\widehat{z}^{\,i}_j = \overline{c}_j \, \chi^i_j \, s^i_j \qquad \forall i \quad \forall j \in \Phi \tag{7.12e}$$

$$\widehat{z}^{\,i} = \sum_{j=1}^{n_t} \widehat{z}^{\,i}_j + \sum_{k=1}^{n_h} \widehat{z}^i_k \qquad \forall i \tag{7.12f}$$

$$v^{i-1}_k + \mathrm{w}^i_k + \textstyle\sum_{j \in \mathcal{H}_k} (y^i_j + p^i_j) = v^i_k + y^i_k + p^i_k \qquad \forall i \quad k = 1 : n_h \tag{7.12g}$$

$$h^i = \frac{\mathrm{g}}{t^i} \sum_{k=1}^{n_h} \rho_k \widehat{\sigma}^i_k y^i_k \qquad \forall i \tag{7.12h}$$

$$\sum_{j=1}^{n_t} g^i_j + h^i = a^i \qquad \forall i \tag{7.12i}$$

$$(1 - \delta_{l2s}) x^1_j \le \sum_{i=1}^{n_i} t^i g^i_j \le (1 + \delta_{l2s}) x^1_j \qquad j = 1 : n_t \tag{7.12j}$$

$$u^i_j + \sum_{k=i}^{\min(i+n_{\mathrm{on}j}, n_i)} d^k_j \le 1 \qquad \forall i \quad \forall j \in \Phi \tag{7.12k}$$

$$d^i_j + \sum_{k=i+1}^{\min(i+n_{\mathrm{off}j}, n_i)} u^k_j \le 1 \qquad \forall i \quad \forall j \in \Phi \tag{7.12l}$$

$$s^i_j - s^{i-1}_j - u^i_j + d^i_j = 0 \qquad \forall i \quad \forall j \in \Phi \tag{7.12m}$$

$$\underline{c}_j \, s^i_j \le g^i_j \le \overline{c}_j \, s^i_j \qquad \forall i \quad \forall j \in \Phi \tag{7.12n}$$

$$\underline{c}_j \le g^i_j \le \overline{c}_j \qquad \forall i \quad j = 1 : n_t, \ j \notin \Phi \tag{7.12o}$$

$$g^i_j \ge \widehat{z}^{\,i}_j \qquad \forall i \quad j = 1 : n_t \tag{7.12p}$$

$$s^i_j, u^i_j, d^i_j \in \mathbb{B} \qquad \forall i \quad \forall j \in \Phi \tag{7.12q}$$

$$0 \le y^i_k \le \overline{y}_k, \ 0 \le p^i_k \le \overline{p}_k, \ 0 \le v^i_k \le \overline{v}_k \qquad \forall i \quad k = 1 : n_h \tag{7.12r}$$

where the variables of the model are $s^i_j, d^i_j, u^i_j, \widehat{z}^{\,i}_j, g^i_j, y^i_k, p^i_k, v^i_k$ and the model is a mixed-integer linear programming problem.

Equation (7.12b) is the definition (7.6) of the market price for each interval using a linearization of the market price function.

Equations (7.12c – 7.12f) are the definition of the zero-priced bids as explained in subsection 7.1.3. Note that the units whose commitment is optimized are multiplied by the state of the unit, s^i_j.

Equations (7.12g – 7.12h) are the water balance in each reservoir and the total hydro generation in interval i, as developed in subsection 7.1.1.

The generation and the demand is always balanced as expressed by equation (7.12i).

Equation (7.12j) links the long term results to the short-term operation. The following information is implicit in these constraints:

- The market-share of our SGC was considered in the long-term planning model, which is where market shares will be taken into account.

- The effect of hydro generation or any type of thermal generation whose fuel availability is subject to a long-term constraint.

  It should be noted that in the case of pseudo-units corresponding to hydro-generation basins of competing companies x_j^1 will be the production of these basins over the whole short-term period. As regards the SGC hydro-production, the long-term planning model may have detailed each reservoir [45], and long-term and short-term models are coordinated via the final volumes in reservoirs $v_k^{n_i}\ \forall\, k$, data which was obtained as a result of the long-term planning solution.

Equations (7.12k – 7.12m) express the unit commitment constraints (see subsection 7.1.4).

Finally, the thermal generation is bounded by the unit minimum and maximum capacity (equations (7.12n) and (7.12o)) and must be equal to at least the zero-priced bid (7.12p). The variables of the unit commitment formulation are binary (7.12q) and the hydro variables must stay between bounds (7.12r).

7.2.2 Nonlinear scheduling model

The nonlinearities considered in this model are derived from the water head function and hydro-generation function in the reservoirs of the SGC (7.1, 7.3).

The zero-priced energy bids $\widehat{z}^i$ obtained from the solution of the linearized unit commitment model are here taken as a parameter, and so is the availability of units and pseudo-units determined by the optimal scheduling, s_j^i. Let $\mathcal{U}^i$ be the set of available thermal units in interval i, which is the set of thermal units that has not been shut down in this interval.

The hourly supply-bid functions employed (7.7) are also nonlinear. However, they become linear for the optimization solver given that we use the zero-priced bids as known parameters.

The complete nonlinear continuous problem to be solved is

$$\max \sum_{i=1}^{n_i} \{a^i \widetilde{m}^{\,i} - \sum_{j\in\mathcal{U}^i} f_j\, g_j^i\} \tag{7.13a}$$

$$\text{s.t. } \widetilde{m}^{\,i} = \{\beta_0^i + \sum_{j\in\mathcal{U}^i} \beta_j^i (g_j^i - \widehat{z}_j^i)\}(a^i - \widehat{z}^{\,i}) + \mathit{\Gamma}^i \qquad \forall\, i \tag{7.13b}$$

$$-\epsilon_\beta \le \sum_{i=1}^{n_i} \Big\{ \widetilde{l}^i - \beta_0^i - \sum_{j\in\mathcal{U}^i} \beta_j^i (g_j^i - \widehat{z}_j^i) \Big\} \le \epsilon_\beta \tag{7.13c}$$

$$v_k^{i-1} + \mathrm{w}_k^i + \sum_{j\in\mathcal{H}_k} (y_j^i + p_j^i) = v_k^i + y_k^i + p_k^i \qquad \forall\, i \quad k = 1 : n_h \tag{7.13d}$$

$$\begin{aligned} h^i = \frac{\mathrm{g}}{t^i} \sum_{k=1}^{n_h} \rho_k \Big\{ & \sigma_{\mathrm{b}k} + \tfrac{\sigma_{\mathrm{l}k}}{2}(v_k^{i-1} + v_k^i) + \\ & \tfrac{\sigma_{\mathrm{q}k}}{3}(v_k^i - v_k^{i-1})^2 + \sigma_{\mathrm{q}k} v_k^{i-1} v_k^i + \\ & \tfrac{\sigma_{\mathrm{c}k}}{4}({v_k^{i-1}}^2 + {v_k^i}^2)(v_k^{i-1} + v_k^i) \Big\} y_k^i \end{aligned} \qquad \forall\, i \tag{7.13e}$$

$$\sum_{j\in\mathcal{U}^i} g_j^i + h^i = a^i \qquad \forall\, i \tag{7.13f}$$

$$(1 - \delta_{l2s}) x_j^1 \le \sum_{i=1}^{n_i} t^i g_j^i \le (1 + \delta_{l2s}) x_j^1 \qquad j = 1 : n_t \tag{7.13g}$$

$$\underline{c}_j \le g_j^i \le \overline{c}_j \quad g_j^i \ge \widehat{z}_j^i \qquad \forall\, i \quad \forall\, j \in \mathcal{U}^i \tag{7.13h}$$

$$g_j^i = 0 \qquad \forall\, i \quad \forall\, j \notin \mathcal{U}^i \tag{7.13i}$$

$$0 \le y_k^i \le \overline{y}_k \quad 0 \le p_k^i \le \overline{p}_k \quad 0 \le v_k^i \le \overline{v}_k \qquad \forall\, i \quad k = 1 : n_h \tag{7.13j}$$

The variables of the model (7.13) are $g_j^i, y_k^i, p_k^i, v_k^i$. From its solution, a market-price estimation $\widetilde{m}^{\,i}$ and the thermal and hydro generations are obtained. In the objective function (7.13a) a nonlinear cost function for the thermal units could be employed instead of the linear terms $f_j\, g_j^i$.

The observations concerning the linearized model (7.12) also apply to the nonlinear one. The main differences are that no zero-priced bids are optimized, that no unit commitment constraints appear, and that the estimated market-price is computed from a four-degree polynomial. Also, a non-linear model for the hydro generation (7.13d – 7.13e) is employed.

7.3 The three-stage procedure

The proposed short-term power planning procedure addresses the development of the bidding strategy of a generation utility participating in a competitive market for a weekly or shorter period. The successive stages of the solution procedure are:

1st stage: Data analysis and preparation

Using records of energy bids and loads from preceding weeks and from similar weekly periods in former years:

- Analyze the type of contribution made by each defined unit to the bids at zero price (either as a given proportion of its power capacity, or as a given proportion of the pool hourly load).
- Fit a polynomial to the forecast of each hourly supply-bid function.
- Determine the relative influence of the amount of power bid made by each unit in the slope of the hourly supply-bid function.
- Forecast the hourly load.

Two of the criteria employed to determine the units that are merged into a pseudo-unit are the type of unit and the uniformity of the coefficients of contribution to the zero-priced energy.

2nd stage: Linearized unit commitment

Find the solution to the unit commitment problem (7.12) using:

- a linearized market-price estimate as in (7.6), where b^i and l^i are estimated in the data analysis stage, but $\widehat{z}^i$ value is optimized and calculated from the unit commitment of this stage, and
- a linearized hydro-generation function (7.4) expressed in terms of the reservoir discharges for the reservoirs of the SGC.

The information obtained from the unit commitment stage is:

- the unit commitment of all units considered,
- the hourly zero-priced energy bids of all units and their sum $\widehat{z}^i$, and
- an initial estimate of hourly market prices $\widehat{m}^i$ and hourly generations.

3rd stage: Nonlinear scheduling refinement

Find a refined solution to the hourly generations using:

- the unit commitment and the hourly zero-priced bids of the committed units or pseudo-units $\widehat{z}^i$,
- a polynomial supply-bid function for each hour with a linear coefficient having a limited range of variation in terms of the generation of the committed units, and
- the nonlinear hydro-generation functions for the reservoirs (7.3) of the SGC.

Note that the zero-priced bids parameter, $\widehat{z}^{\,i}$, in equation (7.7) is here a constant, because it was obtained from the solution of the linearized unit commitment problem. The third stage refines:

- the hourly generation of each unit and the hourly power bids, and
- the estimates of the hourly market prices $\widetilde{m}^i$ and the expected profit.

Ideally, the second and third stages could be combined into a single stage in which a mixed-integer nonlinearly-constrained nonlinear problem would be solved. To date, efficient practical methods for solving such a problem are not available, so the problem is split into two separate stages, the first solving a mixed-integer linear version of the problem, and the second optimizing a continuous nonlinear scheduling refinement in which the mixed-integer solution results are employed as data.

7.4 Results

The preparation of a generation bid for an SGC from the results of the linearized unit commitment and from the results of the nonlinear scheduling is simple. The results of the linearized unit commitment (7.12) indicate which units of the SGC will be operational and when. From (7.12c – 7.12e) and the optimized s_j^i we can readily establish the zero-priced bids $\widehat{z}_j^i$ of each unit at all intervals.

Taking the optimal generations g_j^i and the market prices $\widetilde{m}^{\,i}$ of the nonlinear scheduling solution of (7.13), the generation segment of thermal unit j in interval i between $\widehat{z}_j^i$ and g_j^i should be bid at a price safely below $\widetilde{m}^{\,i}$ (possibly subdivided into sections at increasing price). The power of this unit greater than g_j^i, the optimal expected generation, should be bid at prices safely above $\widetilde{m}^{\,i}$.

Detailed computational results obtained using this procedure can be found in [47].

Chapter 8

Approach to the solution of the long-term planning problem with maintenances

This chapter presents the way we approach the solution of the extension of the LTGP problem with the optimization of the scheduling of the units' maintenance, called the LTGMP problem. The problem formulation is presented in chapter 2, section 2.10.

The LTGMP problem is a quadratic mixed binary problem (QMBP). The standard method for solving this type of problems is branch and bound (B&B, for short).

Branch and bound is an implicit enumeration method for solving pure- or mixed-integer optimization problems. It breaks the problem down into smaller and easier-to-solve subproblems, where the binarity of some variables is relaxed and 0 or 1 values are given to others. This part is called *branching*. Another tool is *bounding*, which is a fast way of finding upper and lower bounds for the optimal solution within a feasible subregion. Through bounding, the enumeration of many combinations of binary variables is avoided.

Basic steps of the B&B algorithm

- Find a binary feasible solution (problem lower bound).
- Initialize the list of unsolved problems: T = the relaxed original problem. Let the problem upper bound be its solution.
- Choose a problem P from the list:
  - × if the upper bound of P is not better than the best feasible solution found → prune by bound, otherwise
  - × solve problem P:
    - ∗ if the solution is unbounded → the associated QMBP is unbounded.
    - ∗ if the solution is infeasible → the associated QMBP is infeasible.
    - ∗ if the solution is feasible for the QMBP → the solution is optimal for the associated QMBP. If necessary, update the best feasible solution found.
    - ∗ otherwise, branch P into two or more subproblems, and add them to the list T. The solution of P gives the upper bound to its child nodes.
  - × Repeat until the list is empty.

The efficiency of a branch and bound procedure for finding the optimal solution is critically dependent on the ability to find good bounds early in order to prune as many nodes as possible and therefore solve the least number of subproblems. Several rules can be established for branching and bounding the problem. The best options depend on the structure of the binary variables.

In this chapter we present two heuristics for finding an initial feasible solution to the LTGMP problem, and the choices of the B&B algorithm implemented.

Our model was tested on 6 realistic cases from the Spanish power system. The data details are in appendix A.2.

8.1 Solutions based on the use of a commercial B&B solver for the LTGMP problem

The number of binary variables for normal-size LTGMP problems (2.26) and the tightness of the model would permit an efficient solution using a commercial QMBP solver (such as CPLEX) if it were not for the exponential number of inequality LMCs (2.26b).

8.1.1 Solution with the CPLEX B&B

Three of the six test cases prepared are of a moderate size (as regards the number of LMCs). The full model (with all the LMCs and binary variables) was solved using AMPL and the mixed integer solver in CPLEX. The solution is presented in table 8.1.

	Obj. Fun.	time (min)	B&B nodes	B&B int. sol.	1st solution gap	1st solution time(sec)
ltp_wm_14	4 973 517 783	1703	192560	19	0.0001	<10
ltp_wm_15	4 742 391 191	1194	74800	21	0.0002	<10
ltp_wm_16	6 029 400 945	32	4491	18	0.0004	110

Table 8.1: Solution to the full model with CPLEX (10^{-5} mipgap)

The column headed *Obj. Fun.* in table 8.1 shows the objective function. Time is indicated in minutes and in cases ltp_wm_14 and ltp_wm_15 the solution took several hours. The adjacent column shows the number of nodes explored by the CPLEX B&B, *B&B nodes*, and the number of integer solutions found, *B&B int. sol.*. The last column shows the gap of the first feasible solution found by CPLEX and the time taken to search for it. It is remarkable that in case ltp_wm_16 CPLEX requires a lot of time (compared to the other two cases) to find the first feasible solution but then solves the full B&B very fast.

For test cases ltp_wm_17 and ltp_wm_18 the number of LMCs is so large that the memory required to store the nodes of the B&B tree generated by CPLEX soon becomes excessive and the solution process is aborted. In table 8.1 the lengthly CPU time required should also be noted, which is partly due to the management of the large amount of memory used by the LMCs in the B&B nodes.

8.1.2 Heuristic for finding a maintenance schedule: MS1 heuristic

The LTGMP problem now has two sets of constraints that impede the use of a direct solution approach for medium- and large-sized problems. These are the exponential number of inequality LMCs, and the binary variables. One way of finding an initial incumbent solution is to relax the problem by considering the all-one LMCs only. So using a mixed-integer solver (such as CPLEX) is appropriate in our case, because our problems do not have many binary variables, but do have an exponential number of LMCs. Once the mixed-integer solution to the maintenance schedule is obtained, the LMCs are reconsidered and the GP heuristic (described in chapter 4) is employed, with the maintenance periods fixed, to find the objective function value of the incumbent.

MS1 heuristic

i Remove the LMCs from (2.26) except the all-one LMCs.

ii Solve the LMC-relaxed problem with a mixed-integer solver
→ maintenance periods are found.

iii Fix the maintenance periods for the LTGP problem.

iv Solve the problem with the GP heuristic and compute the objective function for the incumbent solution.

The time required for finding the optimal solution to problems with a moderate number of units in maintenance may be large. We use the best solution found: either with an optimality gap of 10^{-4} or after running CPLEX for 300 seconds (of CPU time), whichever comes first.

	Obj. func. LMCs rel {0,1} var	gap wrt CPLEX opt/ best sol	gap wrt LMCs rel [0,1] var	time (sec)
ltp_wm_14	4973438352	0.000016	0.0012	29
ltp_wm_15	4742485988	-0.000020	0.0014	23
ltp_wm_16	6028729602	0.000111	0.0023	8
ltp_wm_17	7055391707	0.000010	0.0023	100
ltp_wm_18	4704905249	-0.000077	0.0013	6
ltp_wm_19	7011851945	-0.000731	0.0039	312

Table 8.2: Solution found with the MS1 heuristic using CPLEX and gap with respect to the best known solution and the integrality relaxation solution.

The objective function of the LTGMP problem with the maintenance scheduling found by the MS1 heuristic (see table 8.1.2) reveals that this solution is as good or even better than the CPLEX solution (cases 14 to 16) or the best known solution (cases 17 to 19, which were obtained with a self-developed code). Unfortunately, when this solution is compared with the solution to the relaxation of the variables and of the LMCs (except the all-one LMCs) the gap is still big and does not reflect the quality of the solution.

8.2 Self-developed B&B solver for the LTGMP problem

Problem LTGMP has two types of decision variables: the generation planning and the maintenance scheduling. The maintenance schedule conditions the generation planning given that when a unit is in maintenance during some intervals it can no longer generate in these. For small instances the B&B procedure could be made to consider all the LMCs, but this would add an unnecessary overload to the solution of the subproblems. Our B&B algorithm solves the subproblems with only the all-one LMC of each interval. The rest of the LMCs are considered when an all-binary solution is found (and therefore a maintenance schedule). When a maintenance schedule is found, the corresponding generation planning is solved by the GP heuristic (introduced in chapter 4), with the maintenances fixed.

Therefore, the B&B algorithm handles the binary variables, considering only the all-one LMCs, in order to find the best maintenance schedule.

8.2.1 Bounding the solution

It is important to find good bounds of the optimal solution. These give an indication of the optimality of any feasible solution and they should avoid the need to solve some branches by pruning by bound.

Upper bound:

The solution of the *relaxed QMBP* gives an upper bound of the original problem.

Lower bound:

Any (all-integer) feasible solution to the problem provides a lower bound for the optimal solution value. An initial feasible solution to the LTGMP problem is found using a heuristic.

Bounds on each node:

The solution of a node produces an upper bound for its child nodes. When the tree grows, the upper bound can be updated by the highest upper bound of the tree, which may well be lower than the current one.

8.2.2 Heuristic MS2 to find an all-binary solution

One of the key points in a branch and bound algorithm is finding a good incumbent early, in order to avoid the waste of effort involved in solving weak solutions.

We introduce a heuristic to find an initial solution to the LTGMP problem. It finds a schedule for the maintenance of the units, referred to here as the MS2 heuristic. Then the generation planning problem with the maintenance periods fixed is solved by the GP heuristic. Heuristic MS2 is an extension of the GP heuristic for the LTGMP problem presented in chapter 4.

The maintenance schedule is related to power generation, because the goal is to maximize overall profit. The MS2 heuristic, which finds a maintenance schedule, builds a loading order in several steps (similar to the GP heuristic procedure) by deciding in which intervals it is preferable to have the units in service. By elimination, the maintenance periods become fixed.

As in the GP heuristic, the goal of the MS2 heuristic is to find an appropriate loading order. When one of the units with maintenance intervals requiring optimization is ordered, the unit in this interval is fixed to be in service. The heuristic stops when all the binary variables are fixed.

The loading order is decided by several relaxed solutions of the full model (2.26). Two sets of constraints are relaxed in problem (2.26):

- the load-matching ones; in each interval, we do not use the complete set of LMCs but a subset.
- the integrality of the binary variables, which are made continuous between 0 and 1.

8.2.3 Notation

The relaxation of problem (2.26), in matrix form, is:

$$
\begin{aligned}
\text{minimize} \quad & h'x + \frac{1}{2}x'Hx & & \\
\text{subject to} \quad & Ax \geq a & & \\
& B^i_{L^i} x^i \leq r^i_{L^i} & \forall i & \\
& Cy = v & & \\
& 0 \leq x^i_j \leq \overline{x}^i_j s^i_j & \forall i \quad \forall j \in \Phi & \\
& 0 \leq x^i_j \leq \overline{x}^i_j & \forall i \quad \forall j \in \Omega \setminus \Phi & \\
& y^i_j = [s^i_j; d^i_j; u^i_j] \in [0,1] & \forall i \quad \forall j \in \Phi &
\end{aligned}
\tag{8.1}
$$

where A and a are the matrix and the right-hand side (*rhs*) of the non load-matching constraints, B and r are the matrix and the rhs of the LMCs. All (relaxed) binary variables are merged into vector y. Constraints involving only (relaxed) binary variables have coefficients C and rhs v. The upper bound of the expected generation of a unit in an interval is conditioned to be the unit in service $(\overline{x}^i_j s^i_j)$.

As in chapter 4, the list of LMCs used is L^i. Each element of L^i is a set of units which defines a unique LMC. Matrix $B^i_{L^i}$ represents the submatrix of LMCs defined in L^i and $r^i_{L^i}$ its rhs.

8.2.4 Steps of the MS2 heuristic

The management of the LMCs is similar to that developed for the GP heuristic. The only difference, in the choice of the considered LMCs, is that in the step-by-step stage only one new LMC is added at each iteration and therefore the number of iterations is now $n_i \times n_u$. Let us recall some of the parameters and sets defined for the GP heuristic:

- $\rho^i_j := x^i_j / \overline{x}^i_j$ is the ratio between the expected generation and its upper bound.
- Ω^i_o is the ordered set with the index of the units.
- ϕ^i is the set of units already ordered.

A new set is used in order to know when the maintenance schedule is completed:

- $\eta \subseteq \Phi$ is the set of units with the maintenance already assigned.

The loading order is generated following three different rules:

- In the initialization stage a solution to problem (8.1) which only contains the all-one LMCs and has all the binary variables relaxed is found.

- In the self-ordering stage, for each interval all LMCs made up only of units at upper bound ($\rho_j^i \simeq 1$) are added to the list L^i. If any of these units at upper bound needs to optimize its maintenance period, its binary variables are fixed to be in service.

- In the step-by-step stage, *only* a unit from an interval is chosen and a new solution is found. This unit is the one with the highest ρ_j^i value out of all intervals and units.

As we are interested in finding a maintenance schedule, if a unit has an all-integer solution in all the intervals, we fix this maintenance pattern for this unit.

Because we are solving the relaxed problem, the built-in preprocessing does not detect situations in which some variables could be fixed as if they were being treated as integers. For this reason, it is necessary to purposely implement the preprocessing which detects this situation. It occurs in the case of the constraints that only have binary variables (in our model these are constraints (2.26k) and (2.26l)).

MS2 heuristic

i Initialization

- $\eta := \varnothing$
- $\phi^i := \varnothing \quad \forall\, i$
- $L^i := \{\Omega^i\} \quad \forall\, i$
- Solve (8.1)

ii Self-ordering

- $\phi^i := \{j \in \Omega \,|\, \rho^i_j \simeq 1\} \quad \forall\, i$
- $L^i := L^i \cup \{\omega \,|\, \forall\, \omega \subseteq \phi^i\} \quad \forall i$
- Fix s^i_j to 1 $\forall j \in \phi^i \cap \Phi,\ \forall\, i$ (and d and u accordingly)
- Solve (8.1)

iii While $(\Phi \setminus \eta) \neq \varnothing$, order step-by-step

- Choose the pair (j, i) such that $\rho^i_j = \max_{\forall l \ \forall k \in \Omega \setminus \phi^l} \rho^l_k$
- $\phi^i := \phi^i \cup \{j\}$
- $L^i := L^i \cup \{\phi^i\}$
- if $j \in \Phi$ then
  * fix s^i_j to 1 (and d and u accordingly)
- if any unit $k \in \Phi \setminus \eta$ has an all-integer solution for its binary variables then
  * fix the maintenance schedule for this unit and carry out preprocessing for the constraints with only binary variables.
  * $\eta := \eta \cup \{k\}$
- Solve (8.1)

8.2.5 Branching strategy

The branching method employed consists in choosing a unit and restricting the intervals in which maintenance of this unit can start. Variable d^i_j models the interval i in which unit j starts the maintenance. For each unit, variable $d^i_j,\ i = 1 : n_i$ is a Special Ordered Set of type 1.

Formally, a *Special Ordered Set of Type 1* (SOS1) is a set of decision variables where one variable, but no more, must have a value of 1. Typically these variables correspond to mutually exclusive decisions, for example, choosing the interval in which to start the maintenance of a unit. Given a set of decision variables, $z \in \delta$, the mathematical expression of a SOS1 is:

$$\sum_{j \in \delta} z_j = 1, \qquad z_j \text{ binary} \tag{8.2}$$

The solution may be found more quickly by exploiting the structure of SOS1 when branching.

The set of subproblems defined with the branch and bound procedure defines a tree structure. A node of the tree, N, consists of the range of intervals, from i_a to i_z, where each unit $j \in \Phi$ can start its maintenance:

$$N = \{(j, i_a, i_z) \quad \forall\, j \in \Phi\}$$

The subproblem generated from node N has the constraint set:

$$\sum_{i=i_a}^{i=i_z} d_j^i = 1 \quad \forall\, (j, i_a, i_z) \in N \tag{8.3}$$

instead of (2.26e). As the root node is the relaxation of the original problem, initially $i_a = 2$ and $i_z = n_i - m_j$ for each unit.

Two forms of branching are used: in two parts and in three parts. Suppose that (j_b, i_b) is the chosen unit and interval:

- When the branching is done in two parts the child nodes are:
  - one that restricts the intervals where $d_{j_b}^i$ must be 1 between intervals i_a and i_b: $\sum_{i=i_a}^{i=i_b} d_{j_b}^i = 1$, and
  - the complementary that restricts the intervals where $d_{j_b}^i$ must be 1 between intervals $i_b + 1$ and i_z: $\sum_{i=i_b+1}^{i=i_z} d_{j_b}^i = 1$

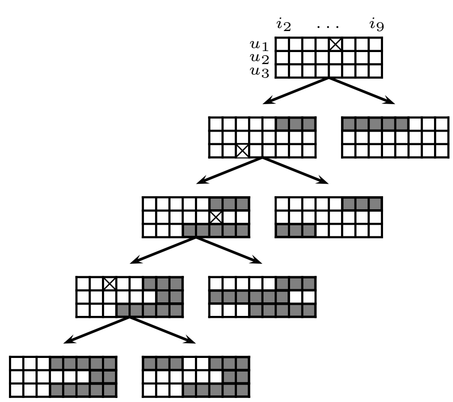


Figure 8.1: Example of branching in two parts (range excluded in shaded cells)

- When the branching is done in three parts the child nodes are:

– one that restricts the intervals where $d^i_{j_b}$ must be 1 between intervals i_a and $i_b - 1$: $\sum_{i=i_a}^{i=i_b-1} d^i_{j_b} = 1$,

– another that fixes the maintenance to interval i_b: $d^{i_b}_{j_b} = 1$, and

– the one that restricts the intervals where $d^i_{j_b}$ must be 1 between intervals $i_b + 1$ and i_z: $\sum_{i=i_b+1}^{i=i_z} d^i_{j_b} = 1$

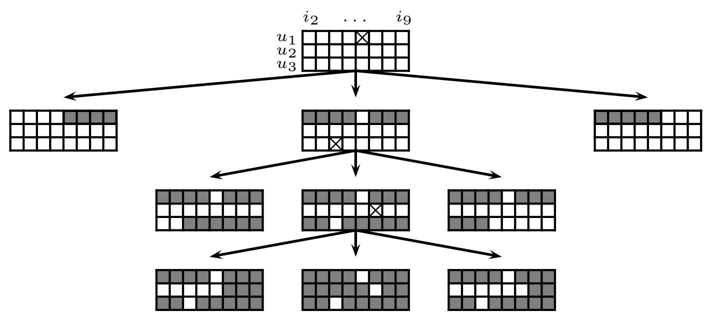


Figure 8.2: Example of branching in three parts (range excluded in shaded cells)

Figures 8.1 and 8.2 show the tree generated from branching in two and in three parts. The tree represents an example of 3 units and 9 intervals, where maintenance can be assigned from the second interval onwards.

8.2.6 Search strategy

The search strategy consists of selecting a child node (similar to a depth-first strategy) until the current node has an all-integer solution or is not solved (because it is infeasible or is pruned).

When branching is carried out in two parts the next node solved is the new left branch. When branching is carried out in three parts the next node solved is the descendant that fixes the maintenance in a single interval. Figures 8.1 and 8.2 represent this scheme.

After one of the branches is solved (because an all-integer solution was found) or discarded (infeasible or pruned) it is necessary to choose a new candidate from the tree. Our choice is the *best-first* option, that is, choosing the node with the best upper bound.

The best-first choice offers a better chance of finding the optimum. The drawback is that the tree grows very quickly. Every so many iterations (for example 2000), we scan the tree for nodes that can be pruned by bound or whose binary variables are already fixed. In the latter case, the generation planning is found with this maintenance schedule fixed.

8.2.7 Variable selection rules

We have implemented four ways of choosing a unit and an interval, (j_b, i_b), for further branching:

(a) Greedy: choose the unit and interval (j_b, i_b) with the largest d value. Branch in three parts.

(b) Balanced: choose the unit and interval (j_b, i_b) where the $\sum_{i=i_a}^{i=i_b} d_{j_b}^i$ is the closest to but smaller than 0.5 and the two sets of intervals with non-zero $d_{j_b}^i$ are furthest apart. Branch in two parts.

(c) d-distribution: We define the distribution of d as $\mathcal{F}_d(i) = \sum_{j\in\zeta} d_j^i,\ i = 1, n_i$, where ζ is the subset of units with their maintenance as yet unfixed.

Every time we choose a branch, and after solving the associated relaxed problem, we store the distribution of d and we save for the solution of the child nodes solved using the depth-first strategy (but not for the sibling branches generated at each step).

First we choose the interval with the highest $\mathcal{F}_d$ value, i_b, and then the unit with the maximum $d_j^{i_b}$. Branch in three parts.

(d) s-distribution: At the beginning of each branch, the s-distribution is computed: $\mathcal{F}_s(i) = \sum_{j\in\zeta} d_j^i,\ i = 1 : n_i$, where ζ is the subset of units with optimized maintenance which are as yet unfixed.

First we choose the interval with the lowest $\mathcal{F}_s$, i_b, and then the unit with the highest $d_j^{i_b}$. Branch in three parts.

Let us take the small example with 3 units and 9 intervals. Assuming that each unit needs 2 intervals for maintenance, suppose that the relaxed solution for d_j^i and s_j^i is:

d_j^i	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9
u_1		0.2			0.8			
u_2	0.1	0.4		0		0.5		
u_3	0.1	0.5		0.2		0.2		
$\mathcal{F}_d$	0.2	1.1	0	0.2	0.8	0.7	0	0

s_j^i	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9
u_1	1	0.8	0.8	1	0.2	0.2	1	1
u_2	0.9	0.5	0.6	1	1	0.5	0.5	1
u_3	0.9	0.4	0.5	0.8	0.8	0.8	0.8	1
$\mathcal{F}_s$	2.8	1.7	1.9	2.8	2	1.5	2.3	3

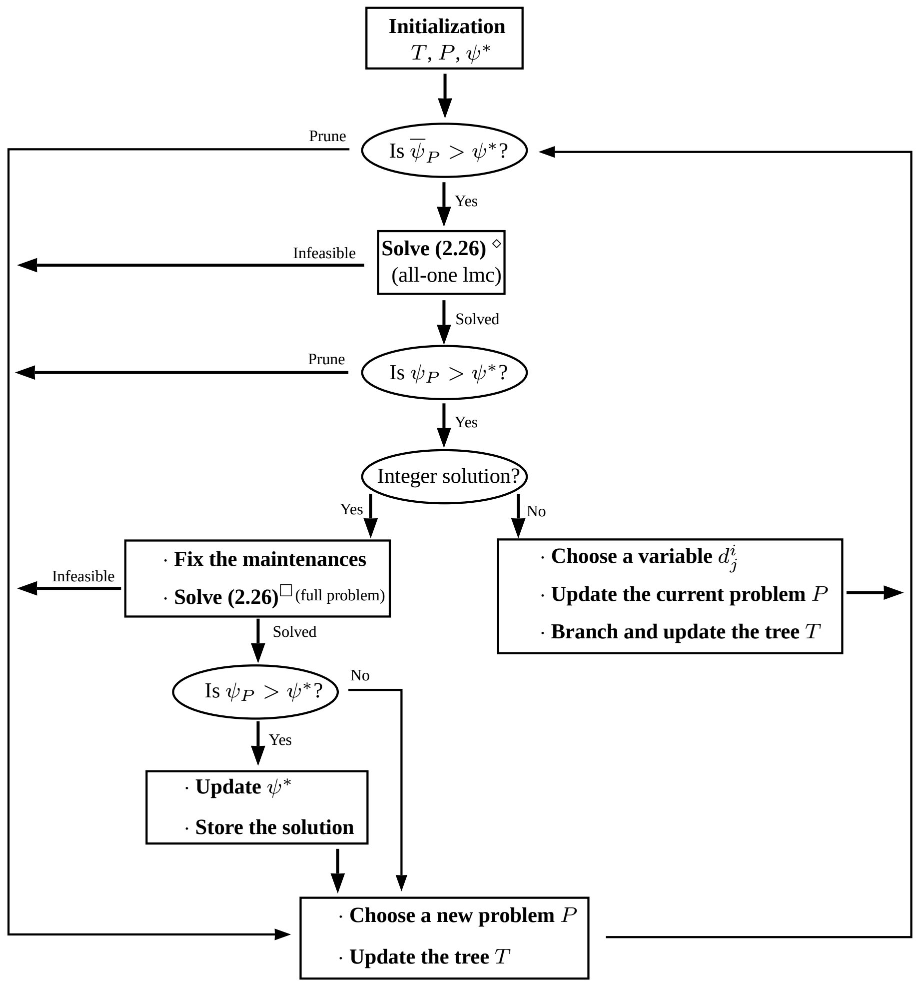
- If strategy (a) is used for branching the unit and interval chosen are (j_1, i_6), because $d_1^6 = 0.8$ and corresponds to the maximum of d.

- Following rule (b) the chosen unit and interval are (j_2, i_5) because the d value of unit 2 is grouped in two sets with $\frac{1}{2}$ on each side. Interval 5 the one that is furthest apart from the two sets.

- The d-distribution sorts the intervals in order to assign the next maintenance. Interval 3 is the first, with $\mathcal{F}_d(3) = 1.1$. Unit 3 is chosen because it has the maximum d value in this interval.

- The s-distribution gives an ordering with the purpose of choosing the intervals. The interval with the minimum $\mathcal{F}_s$ value is i_7, with $\mathcal{F}_s(7) = 1.5$. The chosen unit is j_2 because it has the largest d value in interval 7.

8.2.8 Diagram of the B&B



ψ_P: solution of node P, ψ^*: best objective function found, $\overline{\psi}_P$: objective upper bound of node P.

(2.26)$^\diamond$: Problem (2.26) with the binary variables relaxed and restricted by constraint (8.3) instead of (2.26e), and with only the all-one LMCs.

(2.26)$^\square$: Problem (2.26) with the maintenance periods fixed, which becomes the generation planning problem (solution with the GP heuristic).

8.3 Computational results

8.3.1 Computational results of the self-developed B&B

Table 8.3 presents the results obtained using the four variants of the variable selection rules. The B&B was coded with a script in AMPL and using the quadratic programming solver in CPLEX to solve each subproblem. The LMC right-hand sides are computed when they are required by calling on an external function coded in C++. The B&B is stopped after exploring 10000 nodes. We wish to compare the B&B versions.

		obj. val.	gap	number of nodes prn	inf	int	bet	time (min)
ltp_wm_14	CPLEX	4973517783	0.00001			19		1703
	B&Ba	4973318463	0.00084	292	884	814	6	21
	B&Bb	4971610518	0.00138	483	199	0	-	11
	B&Bc	4973371116	0.00085	1321	822	142	9	14
	B&Bd	4973477329	0.00089	1116	1338	54	5	13
ltp_wm_15	CPLEX	4742391191	0.00001			21		1194
	B&Ba	4742121464	0.00048	2977	642	12	6	12
	B&Bb	4741218029	0.00168	202	487	35	-	10
	B&Bc	4742038747	0.00069	1934	1287	75	10	13
	B&Bd	4741945986	0.00101	1289	1341	32	4	12
ltp_wm_16	CPLEX	6029400945	0.00001			18		32
	B&Ba	6029280293	0.00064	3251	1500	10	3	10
	B&Bb	6027106778	0.00222	723	255	20	-	12
	B&Bc	6028788735	0.00091	2635	1630	107	3	13
	B&Bd	6029137007	0.00099	2487	1711	46	2	12
ltp_wm_17	B&Ba	7054853210	0.00091	3376	656	2	2	13
	B&Bb	7053729817	0.00244	87	387	0	-	12
	B&Bc	7054735929	0.00158	2004	813	35	8	15
	B&Bd	7053900858	0.00204	964	1194	36	1	14
ltp_wm_18	B&Ba	4704545067	0.00040	898	1528	6	3	15
	B&Bb	4704379098	0.00144	213	178	0	-	13
	B&Bc	4704379098	0.00051	1527	576	1	-	14
	B&Bd	4704789226	0.00105	1530	766	40	5	15
ltp_wm_19	B&Ba	7007035892	0.00445	54	738	0	-	19
	B&Bb	7007035892	0.00461	1	186	0	-	18
	B&Bc	7007645618	0.00436	1576	26	2	2	23
	B&Bd	7007035892	0.00442	679	5	0	-	19

Table 8.3: Comparison of solutions found using CPLEX and the four B&B variants implemented with AMPL using the MS2 heuristic (10000 nodes explored)

Within the column headed ”*number of nodes*”, *prn* shows the number of nodes pruned, *inf* is the number of infeasible problems and *int* is the number of all-integer solutions found. The disparity in the number of all-integer solutions found is noteworthy. Nevertheless, the number of solutions that improve the incumbent (best all-integer solution found so far) is small (column *bet*).

On looking at the objective function value of the best solution found (table 8.3), it is not possible to draw clear conclusions about the best strategy. The only strategy that in no case achieves the best solution is the B&Bb. On the other hand, in 5 out of the 6 cases the gap of the solution is the smallest using the B&Ba strategy.

Strategy B&Ba would be chosen because it has the best objective function value in 3 cases and usually delivers better bounds for its solution.

8.3.2 Results using the MS2 heuristic in the self-developed B&B code

Given that the bound found through the MS1 heuristic (using CPLEX and the GP heuristic for the LTGP solutions) is generally better than that found with the MS2 heuristic, we tried substituting the

MS1 heuristic for the MS2 in the self-developed B&B code. The computational results are presented in table 8.4.

				number of nodes				time
		obj. val.	gap	prn	inf	int	bet	(min)
ltp_wm_14	B&Ba	4973438333	0.00081	819	850	418	-	21
	B&Bb	4973438333	0.00101	685	131	0	-	11
	B&Bc	4973453324	0.00083	1743	919	116	1	14
	B&Bd	4973477329	0.00090	1991	1328	44	1	13
ltp_wm_15	B&Ba	4742485988	0.00044	4254	636	0	-	12
	B&Bb	4742485988	0.00140	230	535	0	-	11
	B&Bc	4742485988	0.00065	3090	1090	3	-	12
	B&Bd	4742485988	0.00091	2740	1086	7	-	13
ltp_wm_16	B&Ba	6029280293	0.00088	3868	1280	9	1	10
	B&Bb	6028729499	0.00193	926	259	0	-	11
	B&Bc	6028729499	0.00102	3676	1142	31	-	11
	B&Bd	6029282603	0.00119	3789	1107	7	1	11
ltp_wm_17	B&Ba	7055830944	0.00085	4366	733	0	-	13
	B&Bb	7055830944	0.00214	142	341	0	-	12
	B&Bc	7055830944	0.00148	2849	886	4	-	13
	B&Bd	7055830944	0.00179	1896	1073	11	-	13
ltp_wm_18	B&Ba	4704905227	0.00032	1545	1232	1	-	13
	B&Bb	4704905227	0.00133	225	175	0	-	12
	B&Bc	4704905227	0.00040	2757	721	0	-	13
	B&Bd	4704905227	0.00104	1801	790	17	-	14
ltp_wm_19	B&Ba	7012230408	0.00371	577	302	0	-	19
	B&Bb	7012230408	0.00387	2	185	0	-	18
	B&Bc	7012230408	0.00367	1007	30	0	-	19
	B&Bd	7012230408	0.00367	878	6	0	-	19

Table 8.4: Comparison of the solutions found with the B&B implemented with AMPL using the MS1 heuristic (10000 nodes explored)

From the analysis of the results obtained from the self-implementation of the B&B algorithm with a very good initial incumbent (table 8.4) the following conclusions may be reached:

- In all cases the MS1 heuristic gives better solutions than those obtained with the B&B with the MS2 heuristic.
- The B&B algorithm hardly ever finds all-binary solutions and the initial solution is improved in only a few cases.
- The B&B algorithm reduces the gap by one order of magnitude (compare with table 8.1.2).
- Strategy B&Ba again gives the lowest gap in 5 cases.
- The self-developed B&B was not effective in the largest case ltp_wm_19.

Cases ltp_wm_14 to ltp_wm_19 in tables 8.3 or 8.4 are like cases ltp_14 to ltp_19 of table, for example, 3.1, except in that the maintenance periods in these latter cases were fixed by hand while those of the former were optimized. It may be observed from the objective function values that the overall profits in the optimized-maintenance cases are systematically higher than those in the corresponding cases with no maintenance optimization.

Chapter 9

Conclusions and further research

Conclusions about the LTGP model:

The long-term generation planning for a generation company that participates in a power pool-based market was modeled appropriately as a quadratic programming problem with an exponential number of load-matching constraints. The main modeling features were as follows:

- The Bloom & Gallant formulation was used to model the matching of the generation and the forecasted future demand, represented by a load-duration curve.
- Limits on resource availability or any other operational constraint that can be expressed as a linear combination of the unit expected generation were easily included.
- The development of a market-price function, based on the analysis of the Spanish electricity market, was employed to reward the generation that matches the load-duration curve.
- The market-price function proposed was extended to reflect the influence of overall hydro generation on market prices.

The LTGP model was successfully coupled with a short-term power planning model. This model was also joined with a maintenance scheduling model given that generation planning depends on the maintenance of the units, which was optimized for an SGC.

Conclusions about the short-term generation planning problem:

Short-term generation planning and unit commitment for a generation company which participates in a power pool-based market was addressed. The main modeling aspects were as follows:

- The hourly market price was obtained from two estimates of the supply-bid function.
- Two stages were needed to find the unit commitment and make accurate estimates of power generation.
- It was easy to incorporate the long-term planning results.

The short-term generation planning problem solves two models, one binary with linear (or linearized) objective function and constraints, and the other nonlinear.

Conclusions about the LTGMP model:

The conclusions about the LTGP part of the model also apply here. Given that the maintenance scheduling model is binary, the joint problem results in a quadratic binary mixed problem.

Conclusions about the LTGP problem solution:

The results obtained by implementing the Dantzig-Wolfe column generation implementation show that:

- The variant that tries to reuse the vertices that have been previously generated (DWb) proves to be slower than the one that discards them (DWa), although some vertices are reused several times.
- Convergence near the optimum is very slow, with several hours of computational time needed for medium-size cases.
- The column generation procedures are faster than the active set methods that generate rows.

The results obtained with the GP heuristic show:

- The GP heuristic obtained the optimal solution for all the test cases solved.
- The implementation of the GP heuristic in AMPL with a hook to a routine coded in C++ that computes the LMC right-hand sides proves to be very efficient. The LTGP test cases were solved in 2.2 seconds on average (ranging from 1.01 to 7.31 seconds).
- The implementation of the GP heuristic in C with our self-implementation of the interior-point solver also proves to be very efficient. The LTGP test cases were solved in 1.69 seconds on average (ranging between 0.48 and 6.54 seconds).
- The GP heuristic is much faster than the column generation procedure.

Conclusions about the interior-point solver implemented:

The conclusions are based on our experience with the solution of the LTGP test cases using the heuristic:

- The application in some iterations of multiple-centrality correctors to the Mehrotra predictor-corrector direction saved on average 3 iterations, but there was no significative difference in computational time.

- The application of warm-start techniques to initialize the interior point method in the successive solutions required by the GP heuristic produced a 53% saving in the number of iterations with respect to a cold start. However, there was only a 12 % saving on computational time due to the fact that the greater part of the time was spent on the computation of the LMC right-hand sides.

- Taking an advanced (and well-centered) iterate alone as the initial point for the usual interior-point method reduced the number of iterations by 46% with respect to a cold start, which was only 7% less than when applying more sophisticated warm-start techniques.

Conclusions about the LTGP$_{\mathcal{H}}$ problem solution:

The LTGP model whose objective function depends on the hydro generation was solved with AMPL (the full model) using Minos in the three smallest cases. The use of the GP heuristic with Minos as the solver gave the same results for these small cases and the solution for three additional larger cases proved to be satisfactory.

Conclusions about the LTGMP solution:

The two-part division of the optimization process used to find a LTGMP feasible solution is computationally advantageous, as the problem becomes easier to solve. One part generates maintenance schedules, relaxing some constraints from the generation planning problem, and the other computes generation planning with the maintenances fixed.

The comparison between the three approaches implemented reveals that:

- The maintenance schedule produced by the MS1 heuristic proves to give very good results.

- The MS2 heuristic produces good maintenance schedules but they are inferior to the MS1 ones.

- The self-implementation of the branch and bound algorithm may be considered as a generator of maintenance schedules. When initializing it with the MS2 heuristic, it needs further refinement in order to be competitive with the CPLEX solver.

- The self-implementation of the branch and bound algorithm using the solution given by the MS1 heuristic as an initial solution helps to reduce the solution gap in some measure, but it is unable to find better solutions in most of the cases.

Conclusions about the long- to short-term coordination results:

The solution of the short-term planning model with constraints that limit the generation of the units gives reasonable results, proving that the results of the long-term planning provide valuable information.

9.1 Publications and presentations generated by this thesis

- 2003, joint work with N. Nabona: *Models and Optimization Techniques in Long-Term Electric Power Planning using the Bloom and Gallant Formulation.* Presented in the 21st *TC7 IFIP* conference (Nice, July 2003).

- 2004, joint work with N. Nabona: *Long-term electric power planning in liberalized markets using the Bloom and Gallant formulation.* Chapter in the book *Optimización bajo Incertidumbre*, edited by A. Alonso-Ayuso *et al.* [46]. The contents of this chapter are partially reproduced in chapters 2 and 3 of this thesis.

- 2005, joint work with N. Nabona: *Column Generation Methods and a Heuristic for Solving the Long-term Electric Power Planning Problem.* Presentation in the 8th *SIAM Conference on Optimization* (Stockholm, May 2005).

- 2005, joint work with N. Nabona and J. Gondzio: *Interior point solution for the long term electric power planning problem using the Bloom and Gallant formulation.* Presentation in the 22nd *TC7 IFIP* conference (Torino, July 2005).

- 2006, joint work with N. Nabona: *A heuristic for the long-term electricity generation planning problem using the Bloom and Gallant formulation.* Accepted for publication in the *European Journal of Operational Research* [51]. The contents of this publication are partially reproduced in chapter 4, where further results are presented.

- 2006, joint work with N. Nabona: *A three-stage short-term electric power planning procedure for a generation company in a liberalized market.* Accepted for publication in the *International Journal of Electrical Power & Energy Systems* [47]. The contents of this publication are partially reproduced in chapter 7.

- 2006, joint work with N. Nabona: *A model for the joint optimization of the long-term generation planning and maintenance of the units.* Presented in the *Seminario RETOBI sobre Decisiones bajo incertidumbre en el Sector Eléctrico* (Barcelona, May 2006).

- 2006, joint work with J. Gondzio and N. Nabona: *Warmstarting for interior-point methods applied to the long-term power planning* [52]. Technical report of the Department of Statistics and Operation Research of the Universitat Politècnica de Catalunya (October 2006). The contents of this report are reproduced in chapter 6 of this thesis.

9.2 Topics for further research

9.2.1 Extensions to the long-term generation planning model

Since July 2006, the Spanish electricity market has been extended to include bilateral contracts and Futures. The Futures market is a tool that allows hedging against rising electricity prices in the future. Models to predict the future spot price and the optimal amount of energy to be traded in the Futures market should be included in the long-term generation planning model. The short-term model presented must also be extended in order to take into account the energy agreed in the Futures market or with bilateral contracts.

Given that the electricity can be traded in several markets, some financial precautions should be considered within the long-term planning model by including value-at-risk constraints to avert possible situations of low benefits.

Section 2.12 defines the limits of the model proposed in terms of the stochasticity and the market model employed. A natural extension of the problem is to model hydrogenation availability, basic market prices and fuel and emission allowance costs with scenarios and solve it via Stochastic Programming techniques. Moreover, the LTGP model should incorporate a richer hydro generation representation including reservoir levels and discharges for individual cascaded reservoirs. On the market model side, further analysis must be carried out in order to determine if an equilibrium model would produce more realistic results than the maximum profit used in this thesis.

Power generation from renewable sources, mainly the wind-power generation, is steadily growing in the Spanish power system. Its generation must be considered specifically given that it cannot be scheduled at will. Therefore, a model for the influence and uncertainty of the wind-power generation should be developed and incorporated into the LTGP problem.

9.2.2 Extensions to the solution procedures proposed for solving the models

In this thesis, the LTGP problem was solved with two different techniques: a column-generation one and a heuristic one. The first proved to be slow for real-size problems but it gives the optimal solution. The heuristic procedure is very fast, but although for all the test cases solved always ended in the optimal solution, it might end at a non-optimal point. An interesting topic for further research is to study if there is a direct relationship between the active LMCs for a feasible point and the vertices that define it. Such relationship would relate the solution of the heuristic with a point from which to start a column-generation solution.

By using the properties of the load-matching constraints the size of the largest problem solved by the GP heuristic was of moderate size. However, new instances may require a number of constraints that exceed the available CPU memory. A way of ordering the units in the self-ordering stage should be proposed.

When the long-term generation planning problem considers the effect of the hydro generation on the market price, the objective function becomes indefinite. The interior-point procedure developed should be extended so that it can handle problems with objective functions other than positive-definite quadratics. In the future, it should also solve problems which include some nonlinear constraints.

The joint model of generation planning and maintenance schedule is a quadratic mixed binary problem. A first prototype in AMPL has been presented in this thesis. Further research on the interior-point algorithms may lead to interior-point procedures that speed-up the branch and bound solution of the mixed-integer problems, encountered both in long- and short-term planning. The warm-start procedure may be extended to the cuts generated in the solution of the branch and bound.

Appendix A

Test cases

All the cases were run on a Sunfire V20Z server with 2 processors AMD Opteron 252 at 2.4 GHz and 8 GB of RAM.

All the data used was collected from public web pages (`www.omel.es` and `www.ree.es`). Data and results of all test cases are available at [49].

Detailed data and results for case ltp_14 are in appendix B.

A.1 Cases for testing the LTGP problem

Table A.1 shows the dimensions of the test cases solved: n_u is the number of units, n_i is the number of intervals of which the first is always one-week long, *1st week* is the week of the year where the study begins, the *num. weeks* column shows the period of study of the problem (at least 52 weeks) and n_c lists the number of non-LMCs included in the model. The number of LMCs is also indicated.

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 SGC 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 SGC (e.g., cases ltp_06 and ltp_08) but the competitors are merged into greater or smaller number of units depending on the case. The following (linear) non-load matching constraints were considered:

- the expected hydro generation 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 SGC; and
- special-regime minimum-generation limits.

case	n_u	n_i	1st week	num. weeks	n_c	num. LMCs
ltp_01	13	11	10	94	9	$9.01{\cdot}10^{4}$
ltp_02	15	11	10	94	43	$3.60{\cdot}10^{5}$
ltp_03	17	11	10	94	66	$1.44{\cdot}10^{6}$
ltp_04	18	11	10	94	77	$2.88{\cdot}10^{6}$
ltp_05	45	11	10	94	40	$3.87{\cdot}10^{14}$
ltp_06	63	11	10	94	222	$1.01{\cdot}10^{20}$
ltp_07	18	52	10	52	321	$1.36{\cdot}10^{7}$
ltp_08	25	27	10	53	190	$9.06{\cdot}10^{8}$
ltp_09	52	15	45	59	90	$6.75{\cdot}10^{16}$
ltp_10	29	8	10	52	61	$4.29{\cdot}10^{9}$
ltp_11	33	13	31	52	34	$1.12{\cdot}10^{11}$
ltp_12	67	15	45	59	329	$2.21{\cdot}10^{21}$
ltp_13	56	52	20	53	32	$3.75{\cdot}10^{18}$
ltp_14	13	27	21	53	25	$2.21{\cdot}10^{5}$
ltp_15	14	27	39	53	24	$4.42{\cdot}10^{5}$
ltp_16	12	27	37	53	25	$1.11{\cdot}10^{5}$
ltp_17	16	27	42	53	24	$1.76{\cdot}10^{6}$
ltp_18	18	27	28	53	17	$7.08{\cdot}10^{6}$
ltp_19	19	27	10	53	25	$1.42{\cdot}10^{7}$

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

A.2 Cases for testing the LTGMP problem

Several realistic cases from the Spanish market were prepared. The unit maintenance optimization refers to a specific generation company (SGC).

The coal- and gas-fired units (either single or combined) of the SGC were defined as single units because they are the ones whose maintenance periods are decided by the company. The hydro and nuclear units are grouped into two bigger units. The units from the other participants are merged into several macro-units, depending on the fuel used for producing energy.

These six cases correspond with cases ltp_14 to ltp_19, but optimizing when to start maintenance. The maintenance intervals were assigned randomly in the generation planning problems.

case	n_u	n_i	$\geq$ const	n_m
ltp_wm_14	13	27	25	6
ltp_wm_15	14	27	24	6
ltp_wm_16	12	27	25	5
ltp_wm_17	16	27	24	8
ltp_wm_18	18	27	17	10
ltp_wm_19	34	27	25	21

Table A.2: Test cases information for the LTGMP problem

The number of LMCs is as usual: $n_i(2^{n_u} - 1)$. The number of continuous variables (x_j^i) is $n_i n_u$. The number of binary variables (s_j^i, d_j^i and u_j^i) is $3n_i n_m$.

All the intervals are two-weeks periods except the first one which is always one week.

There are several types of non-LMCs. We considered four: fixed hydro generation over some intervals (equality), maximum nuclear generation (inequality), minimum generation from the special regime (inequality) and a minimum market share for the SGC (inequality).

Appendix B

Full model and solution of case ltp_14

In this appendix the data of the model for case ltp_14 is presented in detail. Following chapter 2, the data used in the basic modeling concepts, the constraints included, the computation of the market price function (with and without the hydro generation influence) for each interval are given. The last section presents the computational results.

B.1 Basic modeling concepts

B.1.1 The long-term horizon

Case ltp_14 finds the generation plan for a horizon of 53 weeks (roughly 1 year) starting in mid May. The first interval is one week and the following intervals are two week long. This produces $n_i = 27$ intervals.

B.1.2 Generation units

This case corresponds to a SGC of medium-to-small size that operates in the Spanish power pool. The units corresponding to the SGC are grouped into 7 units and the units from the rest of the participants are represented by 6 macro units. The problem has $n_u = 13$ units and the set of the units of the pool is defined as $\Omega = \{\mathrm{u}_1^{SGC}, \mathrm{u}_2^{SGC}, \mathrm{u}_3^{SGC}, \mathrm{u}_4^{SGC}, \mathrm{u}_5^{SGC}, \mathrm{u}_6^{SGC}, \mathrm{u}_7^{SGC}, \mathrm{u}_8^{RoP}, \mathrm{u}_9^{RoP}, \mathrm{u}_{10}^{RoP}, \mathrm{u}_{11}^{RoP}, \mathrm{u}_{12}^{RoP}, \mathrm{u}_{13}^{RoP}\}$. The characteristics of the units are displayed in table B.1.

The units belonging to the SGC are marked with the supraindex ^{SGC} and those of the rest of the participants in the pool are marked with the supraindex ^{RoP}. Column headed c_j shows the capacity of unit j (in MW), q_j shows the outage probability of unit j and $\widetilde{f}_j$ the generation cost (in €/MWh). An extra unit should be considered in order to represent the unserved energy by the units in the pool, u_{14}. In practice this unit acts as a dummy variable and only its price is used in the objective function. The price of the external energy is 94.2 €/MWh. The cost used in the models f_j is the difference between the generation cost $\widetilde{f}_j$ and $\widetilde{f}_{14}$.

The next two columns of table B.1 show the maintenance of the units. We have assumed that unit u_8 and unit u_{10} are not available in interval 3 and 6 respectively because of fixed maintenance.

units	c_j (MW)	q_j	f_j (€/MWh)	$\mathcal{M}_j$	m_j	fuel
u_1^{SGC}	753	0.05	35	8, 9	2	fuel/gas
u_2^{SGC}	160	0.01	19	22, 23	2	coal
u_3^{SGC}	160	0.06	21	14, 15	2	coal
u_4^{SGC}	80	0.4	25	18	1	coal
u_5^{SGC}	221	0.2	15	26	1	coal
u_6^{SGC}	324	0.1	17	10	1	coal
u_7^{SGC}	665	0	0			hydro
u_8^{RoP}	10201	0.03	30	3		comb. cycle
u_9^{RoP}	18696	0	0			hydro
u_{10}^{RoP}	5877	0.19	40	6		fuel/gas
u_{11}^{RoP}	7876	0.02	5			nuclear
u_{12}^{RoP}	10620	0.07	20			coal
u_{13}^{RoP}	17112	0.4	60			special reg.

Table B.1: Characteristics of the units of case ltp_14.

Therefore set $\mathcal{M}_8 = \{3\}$, $\mathcal{M}_{10} = \{6\}$ and set $\mathcal{M}_j$ is empty for the rest of units of the rest of participants. This test case was also solved with the optimization of the maintenance (case ltp_wm_14). The thermal units whose maintenance is optimized by the SGC is $\Phi = \{u_1^{SGC}, u_2^{SGC}, u_3^{SGC}, u_4^{SGC}, u_5^{SGC}, u_6^{SGC}\}$. Parameter $m_j \; \forall j \in \Phi$ shows the number of intervals that each unit requires to allow maintenance. When this test case was solved with all the maintenance periods fixed a random maintenance schedule was assigned to each unit. The last column shows the fuel used by each unit.

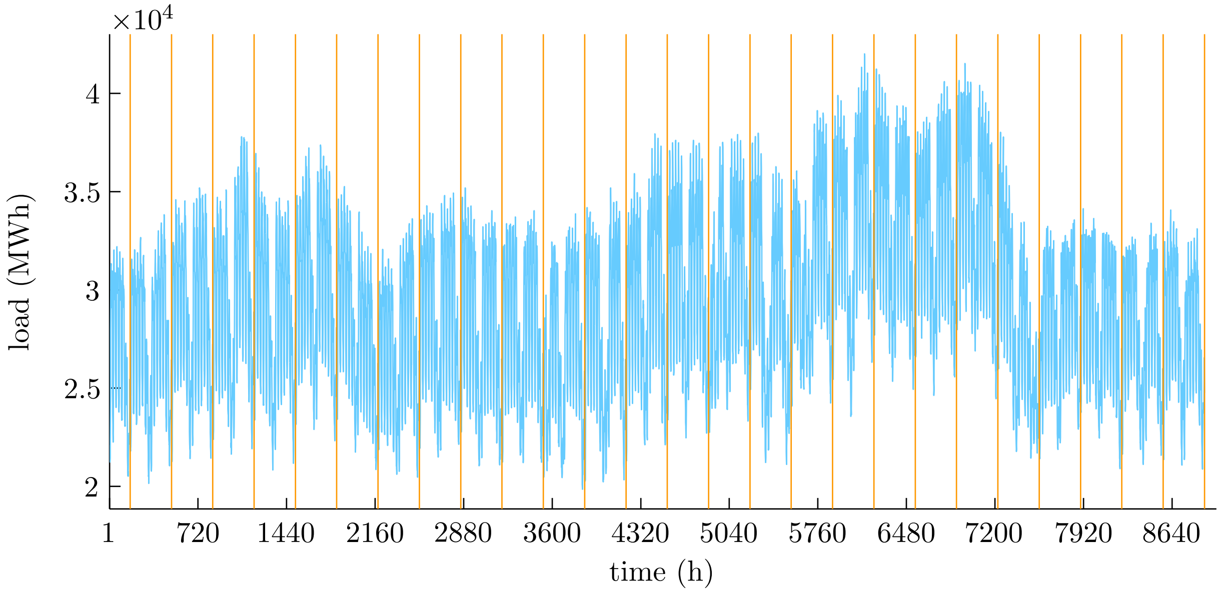
Note that u_7^{SGC} and u_9^{RoP} are hydro units and therefore their outage probability and generation cost is 0. Unit u_{13}^{RoP} represents the units that produce within the special regime rules (such as wind power mills, cogeneration plants or other units which use green sources). This unit has a capacity of 17112 MW (as issued by Red Eléctrica `http://www.ree.es` in its official bulletin). The generation cost is assumed to be high (60 €/MWh) and given that this type of generation is rather unpredictable (specially the wind power generation) the outage probability considered is high (0.4).

The set of indices of the units sorted in merit merit order is: $\Omega_o = \{7, 9, 11, 5, 6, 2, 12, 3, 4, 8, 1, 10, 13\}$. Should the load-matching constraints be the only constraints to satisfy, Ω_o would be the loading order in each interval.

B.1.3 The load-duration curve

The load-duration curve for each of the 27 intervals was computed from the load series that occurred from May 2004 to May 2005 in Spain. The time series of the load is plotted in figure B.1.3. These LDCs rescaled and rotated correspond to the load-survival function $S_\varnothing(z)$ for each interval (see section 2.2.4).

The main characteristics of each LDC are in table B.2.



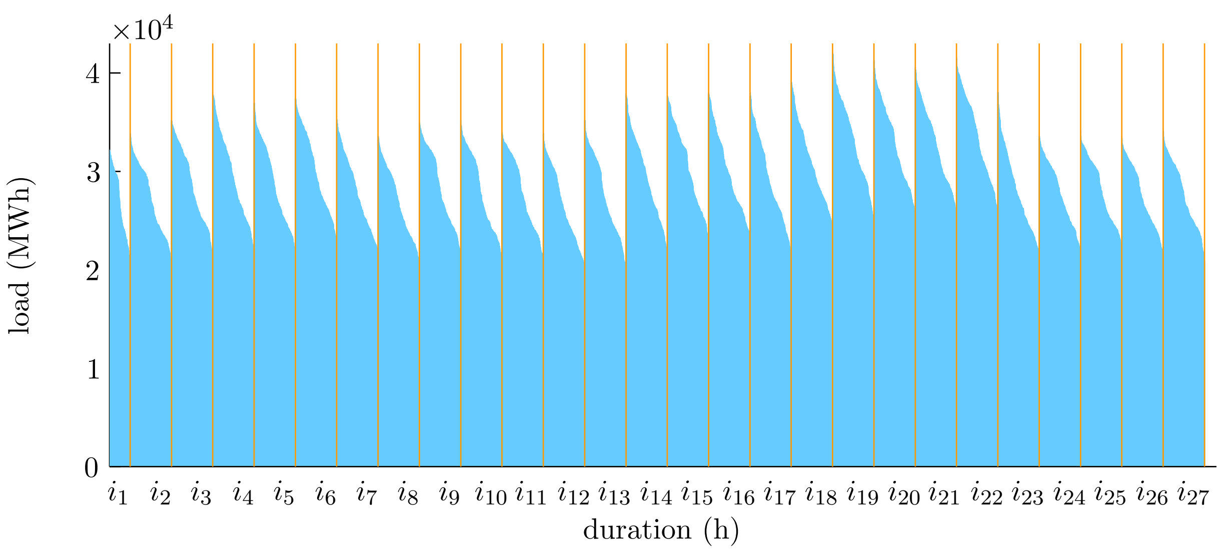


Figure B.1: Time series of the load and corresponding LDC of each interval defined in case ltp_14 (the first interval starts in mid May).

interval	t^i (h)	$\overline{p}^i$ (MW)	$\underline{p}^i$ MW	$\widehat{e}^i$ MW
i_1	168	32201	20519	4492220
i_2	336	33824	20151	9143949
i_3	336	35191	21265	9605066
i_4	336	37789	21642	9890653
i_5	336	36937	20837	9646188
i_6	336	37375	22316	10074722
i_7	336	35263	20851	9371116
i_8	336	33617	20469	9026936
i_9	336	34925	21489	9536476
i_{10}	336	35194	21044	9374434
i_{11}	336	34036	21204	9333632
i_{12}	336	33870	19849	8905714
i_{13}	336	35190	20235	9234470
i_{14}	336	37940	21295	9867892
i_{15}	336	37616	22473	10209783
i_{16}	336	37908	22755	10031221
i_{17}	336	37972	21120	9768858
i_{18}	336	39125	24079	10380705
i_{19}	336	42017	23558	11115242
i_{20}	336	41234	24935	11058906
i_{21}	336	40603	24630	10988671
i_{22}	336	41522	24460	11255993
i_{23}	336	38025	21439	9392329
i_{24}	336	33565	21410	9334503
i_{25}	336	34135	20905	9421309
i_{26}	336	33392	21116	9219790
i_{27}	336	34061	20883	9290338

Table B.2: Information about each LDC of case ltp_14.

B.1.4 Load-matching constraints

Given that there are 16 intervals with 13 units available and 11 intervals with 12 units available the number of load matching constraints is: $16(2^{13} - 1) + 11(2^{12} - 1) = 176101$ LMCs. In case of solving the problem with a direct approach, all the LMC constraints should be generated (both the body and the right-hand side of the constraint). The number of LMCs needed in case of optimizing the maintenance schedule is 212965.

B.1.5 Non-load-matching constraints

Case ltp_14 includes the following constraints:

Maximum hydro generation

The hydro generation of the period under study was bounded with 4 constraints, each of which limits the generation every three months. The sets and parameters that define these constraints (see section 2.7) are:

- The set of successive intervals is $\mathcal{I} = \{\{1,2,3,4,5,6,7\}, \{8,9,10,11,12,13,14\}, \{15,16,17,18,19,20,21\}, \{22,23,24,25,26,27\}\}$.
- The set of hydro units is $\mathcal{H} = \{\mathrm{u}_7^{SGC}, \mathrm{u}_9^{RoP}\}$.
- The right-hand side parameters are:

$$\begin{array}{lcl} \kappa_{\mathcal{H}}^{\{1,2,3,4,5,6,7\}} & = & 7230000 \\ \kappa_{\mathcal{H}}^{\{8,9,10,11,12,13,14\}} & = & 6111000 \\ \kappa_{\mathcal{H}}^{\{15,16,17,18,19,20,21\}} & = & 9170000 \\ \kappa_{\mathcal{H}}^{\{22,23,24,5,26,27\}} & = & 8040000 \end{array}$$

These coefficients were estimated from historical records of hydro generation.

SGC market-share

We assume that our SGC maintains a minimum market share on a monthly basis. The SGC considered in this case wishes to plan its generation to have a share of at least the 5%. The generation of some intervals was grouped in order to compute the SGC market-share. The market-share constraints for a group of intervals were formulated as:

$$\sum_{i \in \iota} \sum_{j \in \Phi} x_j^i \geq \kappa_{ms}^{\iota\, SGC}$$

where ι is a subset of grouped intervals. The coefficients of the right-hand side of the constraint are:

grouped intervals	total load	$\kappa_{ms}^{\iota\, SGC}$
$\{1,2\}$	13636169	681808.46
$\{3,4\}$	19495719	974785.93
$\{5,6\}$	19720911	986045.53
$\{7,8\}$	18398052	919902.58
$\{9,10\}$	18910910	945545.48
$\{11,12\}$	18239345	911967.27
$\{13,14,15\}$	29312145	1465607.27
$\{16,17\}$	19800079	990003.97
$\{18,19\}$	21495947	1074797.33
$\{20,21\}$	22047577	1102378.84
$\{22,23\}$	20623982	1031199.12
$\{24,25\}$	18755812	937790.59
$\{26,27\}$	18510128	925506.40

Minimum generation of the special regime units

We have assumed that the unit in special regime u_{13}^{RoP} has high production costs. In order to model this type of generation appropriately it is necessary to include a constraint that guarantees a minimum generation level. As in the hydro generation constraints the limits are quarterly. Using historical records of generation of this technology we have employed the following constraints:

$$\sum_{\iota \in \mathcal{I}} x_{13}^{RoP} \geq \kappa_{sr}^{\iota} \qquad \forall\, \iota \in \mathcal{I}$$

with right-hand sides:

$$\begin{aligned}
\kappa_{sr}^{\{1,2,3,4,5,6,7\}} &= 9218250 \\
\kappa_{sr}^{\{8,9,10,11,12,13,14\}} &= 10619100 \\
\kappa_{sr}^{\{15,16,17,18,19,20,21\}} &= 11882250 \\
\kappa_{sr}^{\{22,23,24,5,26,27\}} &= 9901350
\end{aligned}$$

Maximum generation of the nuclear units

In this case the nuclear macro unit does not have any maintenance assigned. However, in practice it is likely that some of the components of this macro unit will stop. Using historical data we have limited the generation of the nuclear unit in intervals of three months (like the hydro generation constraint). The constraints employed are:

$$\sum_{\iota \in \mathcal{I}} x_{11}^{RoP} \leq \kappa_{nuc}^{\iota} \qquad \forall\, \iota \in \mathcal{I}$$

with right-hand sides:

$$\begin{aligned}
\kappa_{nuc}^{\{1,2,3,4,5,6,7\}} &= 16477000 \\
\kappa_{nuc}^{\{8,9,10,11,12,13,14\}} &= 16382500 \\
\kappa_{nuc}^{\{15,16,17,18,19,20,21\}} &= 17748500 \\
\kappa_{nuc}^{\{22,23,24,5,26,27\}} &= 14258000
\end{aligned}$$

Other LMCs not included in this case

This case does not have any representation of a unit whose generation with national coal is incentivized. The inclusion of a unit with this characteristic would be modeled as:

- Assume that unit u_k receives κ_{red} €/MWh as a compensation for burning national coal when it is generating. This compensation is received for the firsts $\overline{\kappa}_{red}$ MW produced.
- Unit u_k is modeled as two pseudo-units: $u_{\tilde{k}}$ and $u_{\hat{k}}$, each having the same capacity and outage probability but with different generation costs.
- Two new constraints are required:

- one that limits the production of the incentivized unit (for example $u_{\tilde{k}}$):

$$\sum_{i=1}^{n_i} x^i_{\tilde{k}} \leq \overline{\kappa}_{red}$$

- another that limits the duration of the generation of the two new units:

$$x^i_{\tilde{k}} + x^i_{\hat{k}} \leq t^i(1 - q_k)c_k$$

Another constraint not included in the model is the minimum generation time constraint. This constraint is checked after solving the problem with no such constraints. If a unit appears not to produce the required level in some interval, the problem is enlarged with these new constraints and is re-solved.

B.1.6 Market-price function with respect to load duration

Table B.3 shows the parameters that define the market-price function used in this test case. The first columns show the intercept and slope of the market-price function with respect to the load. The following columns show the parameters of the market-price function fitted taking into account the total hydro generation of the interval (see section 2.8).

	$b^i + l^i t$		$b_0^i - c_0^i \sum_{k\in\mathcal{H}} x_k^i + l^i t$		
interval	b^i	l^i	b_0^i	l^i	c_0^i
i_1	45.92	-0.1479	105.26	-0.1560	$9.062{\cdot}10^{-5}$
i_2	49.50	-0.0868	101.94	-0.0926	$4.222{\cdot}10^{-5}$
i_3	47.22	-0.0797	125.46	-0.1241	$5.829{\cdot}10^{-5}$
i_4	50.72	-0.0893	127.66	-0.1503	$5.466{\cdot}10^{-5}$
i_5	51.78	-0.0928	113.61	-0.1333	$5.348{\cdot}10^{-5}$
i_6	53.85	-0.0993	141.15	-0.1495	$8.608{\cdot}10^{-5}$
i_7	47.51	-0.0773	96.40	-0.1047	$4.934{\cdot}10^{-5}$
i_8	47.09	-0.0751	110.25	-0.0995	$8.081{\cdot}10^{-5}$
i_9	52.95	-0.0951	124.54	-0.1434	$8.088{\cdot}10^{-5}$
i_{10}	57.76	-0.1084	98.42	-0.1244	$4.525{\cdot}10^{-5}$
i_{11}	59.34	-0.1149	91.42	-0.1189	$3.961{\cdot}10^{-5}$
i_{12}	53.26	-0.0958	93.60	-0.0979	$4.772{\cdot}10^{-5}$
i_{13}	54.18	-0.1045	80.89	-0.0864	$2.954{\cdot}10^{-5}$
i_{14}	59.34	-0.1218	91.50	-0.0953	$3.248{\cdot}10^{-5}$
i_{15}	60.75	-0.1189	91.87	-0.1088	$2.103{\cdot}10^{-5}$
i_{16}	59.04	-0.1177	89.68	-0.1079	$1.659{\cdot}10^{-5}$
i_{17}	42.42	-0.0779	82.27	-0.1338	$1.267{\cdot}10^{-5}$
i_{18}	47.06	-0.0879	70.24	-0.1162	$8.140{\cdot}10^{-5}$
i_{19}	52.37	-0.0992	68.49	-0.0851	$1.217{\cdot}10^{-5}$
i_{20}	50.38	-0.0932	73.36	-0.0831	$1.459{\cdot}10^{-5}$
i_{21}	47.89	-0.0790	79.11	-0.1169	$1.435{\cdot}10^{-5}$
i_{22}	46.18	-0.0706	90.29	-0.1091	$1.735{\cdot}10^{-5}$
i_{23}	46.30	-0.0693	78.45	-0.0874	$2.095{\cdot}10^{-5}$
i_{24}	46.92	-0.0667	68.82	-0.0530	$1.998{\cdot}10^{-5}$
i_{25}	44.70	-0.0714	82.24	-0.0549	$3.193{\cdot}10^{-5}$
i_{26}	36.31	-0.0418	72.75	-0.0460	$2.268{\cdot}10^{-5}$
i_{27}	42.03	-0.0645	109.45	-0.0726	$5.016{\cdot}10^{-5}$

Table B.3: Market-price function with respect to load duration (left) and with the influence of the hydro generation level (right). It is used for computing the expected profit of each unit (2.9) or (2.11).

B.2 Solution

B.2.1 LTGP problem solution

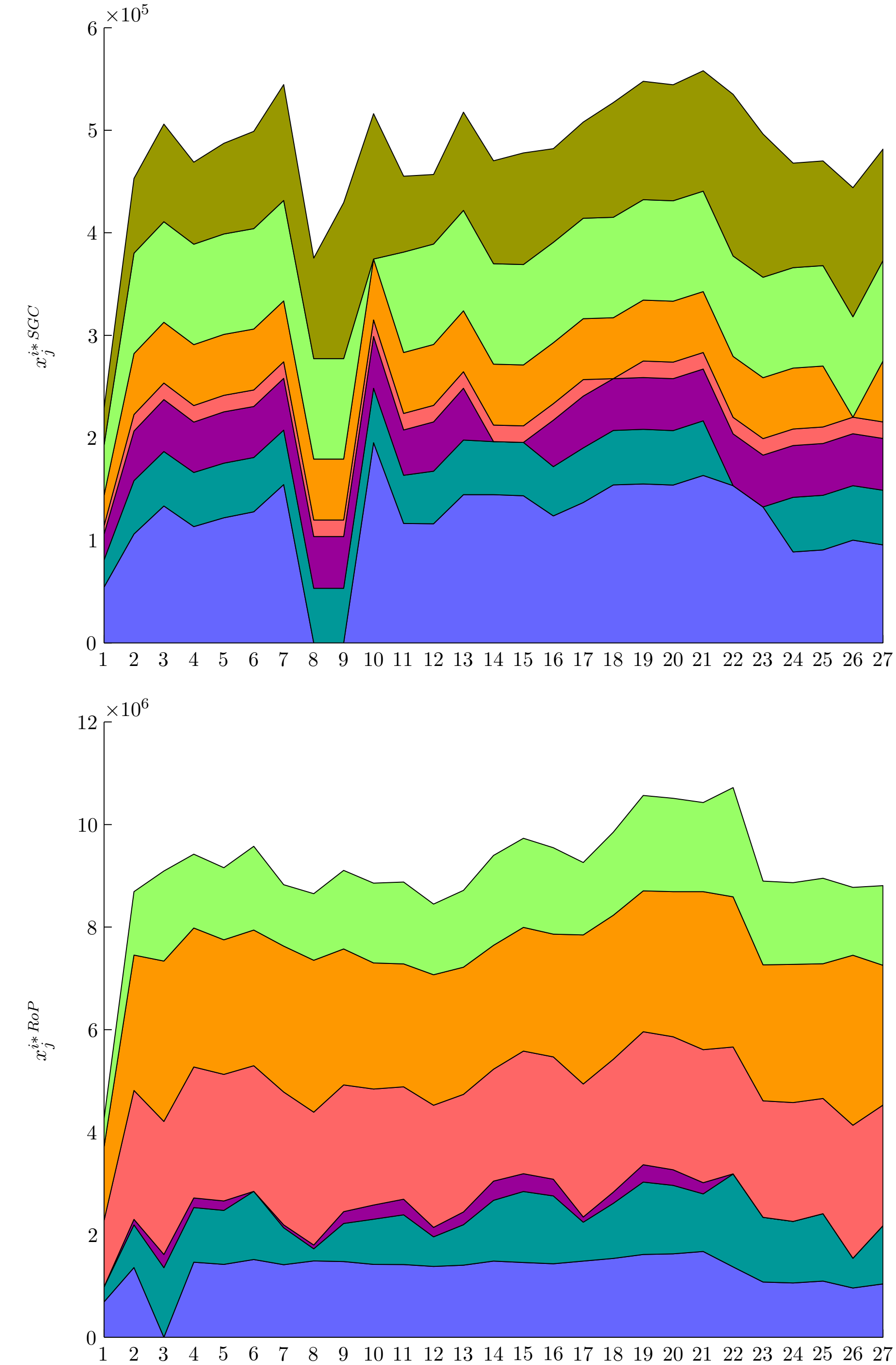


Figure B.2: Optimal solution plan after solving model LTGP. The graphic above contains the units of the SGC and the graphic below contains the units of the RoP, both stacked in natural order.

B.2.2 LTGP$_{\mathcal{H}}$ problem solution

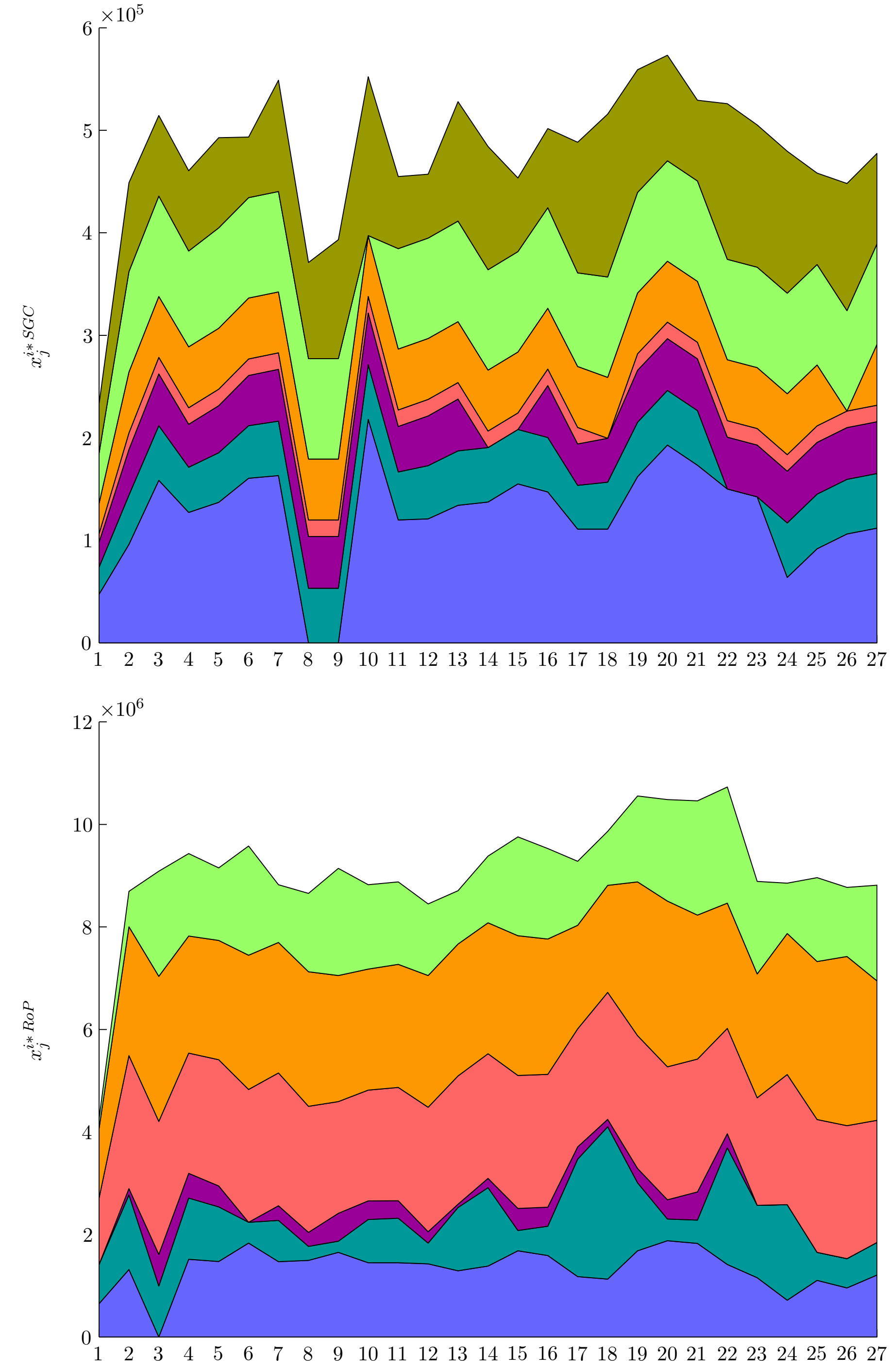


Figure B.3: Optimal solution plan after solving model LTGP$_{\mathcal{H}}$. The graphic above contains the units of the SGC and the graphic below contains the units of the RoP, both stacked in natural order.

B.2.3 LTGMP problem solution

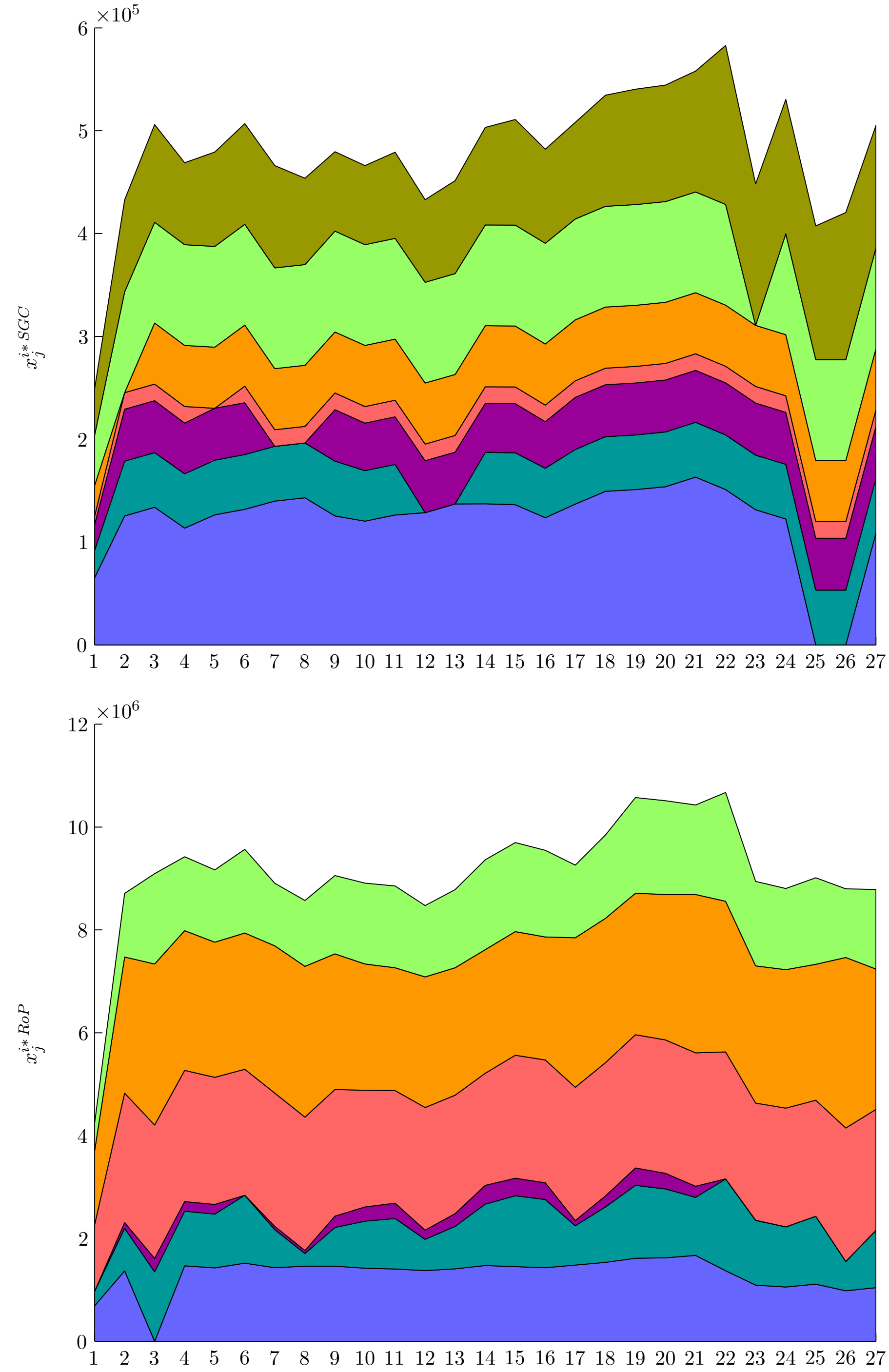


Figure B.4: Optimal solution plan after solving model LTGMP. The graphic above contains the units of the SGC and the graphic below contains the units of the RoP, both stacked in natural order.

unit	random schedule	optimized schedule
u_1^{SGC}	8, 9	25,26
u_2^{SGC}	22, 23	12,13
u_3^{SGC}	14, 15	7,8
u_4^{SGC}	18	5
u_5^{SGC}	26	2
u_6^{SGC}	10	23

Table B.4: Intervals in which the units are fixed in maintenance for models LTGP and $LTGP_{\mathcal{H}}$ (random schedule) and maintenance schedule that results from solving model LTGMP.

B.2.4 Comments on the results

The solution presented for each of the models is separated into two graphics, one for the SGC and another for the RoP units. The reason is the different magnitude of its value (note that the graphic with the SGC solution is scaled by 10^5 and the graphic with the RoP solution is scaled by 10^6).

In each graphic of the SGC, hydro unit u_7^{SGC} corresponds to the top area stacked. In each graphic of the RoP, hydro unit u_9^{RoP} corresponds to the second from the bottom area stacked.

As it can be observed in the solution of the $LTGP_{\mathcal{H}}$ model (graphic B.3), the hydro generation of all the units is redistributed strategically by increasing the production in intervals 17 and 18.

As regards the optimization of the maintenance schedule, table B.4 shows a total change of maintenance intervals with respect to the random maintenance intervals first employed. With this change the objective function is improved from 4970958830 to 4973438333.

Appendix C

Glossary of symbols

Abbreviations

B&B	Branch and bound (algorithm).
ETS	Emission trading scheme.
GCD	Generation duration curve.
GP	Generation planning (heuristic).
LDC	Load-duration curve.
LMC	Load-matching constraint.
LTGMP	Long-term generation and maintenance planning (problem).
LTGP	Long-term generation planning (problem).
$\text{LTGP}_{\mathcal{H}}$	LTGP with changes in market-price function due to hydro generation level.
MIBEL	Iberian (Portuguese + Spanish) electricity market.
MO	Market operator.
M&S	Murtagh & Saunders (algorithm).
QMBP	Quadratic mixed binary problem.
rhs	Right-hand side (of a constraint).
RoP	Rest of participants with respect to a SGC in a power pool.
SGC	Specific generation company participating in a power pool.
SO	System operator.

Indices

$_B$	Subscript that refers to the set of basic variables.
$_d$	Subscript that refers to the dual variables.
^{i}	Superscript that indicates the interval.
^{Gc}	Superscript that refers to the Gondzio centrality-corrector step.
$_j$	Subscript that indicates the generation unit.

$_L$	Subscript that refers to the load-matching constraints.
^{Mc}	Superscript that refers to the Mehrotra corrector step.
^{Mp}	Superscript that refers to the Mehrotra predictor step.
$_N$	Subscript that refers to the set of non-basic variables.
$_{nL}$	Subscript that refers to the non-load-matching constraints.
^{P}	Superscript for a general predictor step.
$_p$	Subscript that refers to the primal variables.
$_S$	Subscript that refers to the set of superbasic variables.
$_{ub}$	Subscript that refers to the upper bound constraints.
$_\mu$	Subscript that refers to the centrality.
$\tilde{}$	Superscript that indicates new variables or constraints (except in chapter 7).
$'$	Indicates the transpose.

Dimensions

n_B	Number of basic variables.
n_c	Number of non-load-matching constraints.
n_h	Number of hydro reservoirs.
n_i	Number of intervals.
n_l	Number of units not perfectly ordered (forming a landing).
n_m	Number of units that optimize their maintenance schedule ($n_m = \|\Phi\|$).
n_N	Number of nonbasic variables.
n_s	Number of constraints that involve only binary variables.
n_S	Number of superbasic variables.
n_t	Number of thermal units.
n_u	Number of units ($n_u = \|\Omega\|$).
n_u^i	Number of units available in interval i.
n_V	Number of vertices.

Parameters

A	Matrix of non-LMCs: $A \in \mathbb{R}^{n_c \times n_u \cdot n_i}$.
a	Right-hand side vector of the non-LMCs (A matrix): $a \in \mathbb{R}^{n_c}$.
a^i	Load to be matched in (hourly) interval i.
B	Matrix of LMCs: $B \in \mathbb{R}^{(2^{n_u}-1) \cdot n_i \times n_u \cdot n_i}$.
b^i	Chap. 2: Intercept of the market-price function with respect to load-duration in interval i.

	Chap. 7: Intercept of the linearized supply-bid function in interval (hour) i.
b_0^i	Intercept of the market-price function with correlation with the hydro generation.
$\mathscr{C}$	Central path.
c_j	Capacity of unit j (in MW).
$\overline{c}_j$	Maximum power capacity of unit j.
$\underline{c}_j$	Minimum power capacity of unit j.
c_0^i	Correlation coefficient of the hydro generation with the market-price function.
e	All-one vector.
$\widehat{e}^{\,i}$	Total energy in the LDC of interval i.
f_j	Linear generation cost of unit j (€/MWh).
$f_{\mathrm{su}j}$	Start-up cost of unit j.
h	Vector for the linear part of the objective function ψ.
H	Matrix of the quadratic part of the objective function ψ.
I	Identity matrix.
l^i	Chap. 2: Slope of the market-price function with respect to load-duration in interval i.
	Chap. 7: Slope of the linearized supply-bid function in (hourly) interval i.
$\widetilde{l}^i$	Slope of the non-linear supply-bid function in (hourly) interval i.
m_j	Number of intervals that unit j needs for maintenance ($j \in \Phi$).
M	Matrix of general active constraints.
$\widehat{M}$	Matrix of active constraints including the active bounds.
$n_{\mathrm{on}j}$	Minimum number of (hourly) intervals that unit j must be in operation.
$n_{\mathrm{off}j}$	Minimum number of (hourly) intervals that unit j must be idle.
$n_u + 1$	Index that represents the external energy.
$\overline{p}^{\,i}$	Peak load of the LDC in interval i.
$\underline{p}^{\,i}$	Base load of the LDC in interval i.
$\widehat{p}_c$	SGC capacity.
q_j	Probability of random failure of unit j.
r	Right-hand side vector of the LMCs (B matrix): $r \in \mathbb{R}^{(2^{n_u}-1)\cdot n_i}$.
t^i	Time duration (in hours) of interval i.
V	Set of vertices defined by the LMCs in each interval.
w_k^i	Natural inflow in reservoir k during interval i.
$\overline{x}_j^i$	Upper bound of the expected generation of unit j during interval i.
Z	Matrix whose columns form a basis for the set of vectors orthogonal to the rows of the linear constraint matrix.

$\mathbb{0}$	Matrix of zeros.
β	Scaling factor (weight) of a direction.
β_0^i, β_j^i	Weights to the contribution of each unit to the variation of $\widetilde{l}^i$.
$\underline{\beta}$	Lower relative threshold for outlier complementarity products.
$\overline{\beta}$	Upper relative threshold for outlier complementarity products.
δ_c	Value close to but strictly lower than 1.
δ_{l2s}	Tolerance with which the short-term solution is allowed to differ from the long-term results.
γ	Value used for building the interval in which the complementarity products must lay.
γ_j	Coefficient which relates the energy generated by unit j with the CO_2 emissions.
γ_q, γ_c, γ_t	Coefficients of the non-linear supply-bid function.
ϵ_g	Small tolerance value, such as 10^{-6}.
ϵ_b	Tolerance for accepting a point as feasible.
ϵ_c	Infeasibility tolerance below which an iterate is stored as a warm start point.
ϵ_h	Small value used to increase a hypothetical stepsize.
ϵ_s	Minimum step size along the Newton direction.
ϵ_w	Maximum infeasibility allowed in order to not perform any warm-start iteration.
ϵ_β	Variation allowed to the linear coefficient $\widetilde{l}^i$ in order to avoid collusion.
η	Chap 3: Eta matrix (identity matrix except in one column).
κ	New target for the reduction of the primal and dual infeasibility.
κ_c	Minimum SGC capacity ratio (for the units that optimize the maintenance schedule).
$\kappa_{CO_2}^{RoP}$	CO_2 allowance allocated to the RoP.
$\kappa_{CO_2}^{SGC}$	CO_2 allowance allocated to the SGC.
$\kappa_{\mathcal{H}}$	Right-hand side of the expected hydro generation constraint.
κ_s	Minimum system capacity ratio.
λ_k^i	Chap. 7: Proportion of zero-priced energy bid of reservoir k in interval i with respect to the demand.
μ	Duality measure at a given point.
ν	Chap. 5: Target of the new complementarity products.
ξ	Vector of infeasibility of the constraints.
χ_j^i	Proportion of zero-priced energy bid of unit j with respect to the unit maximum capacity.
ρ_j^i	Chap. 4 and 8: Ratio between the expected generation variable (x_j^i) and its upper bound ($\overline{x}_j^i$).
ρ_k	Chap. 7: Efficiency of the turbine-alternator system in a hydro reservoir.
ϱ	Chap. 2: Expected price for emission allowances at the EU ETS.

σ	Chap. 5 and 6: Centering parameter of the Newton search direction.
$\sigma_{\mathrm{b}k}$, $\sigma_{\mathrm{l}k}$, $\sigma_{\mathrm{q}k}$, $\sigma_{\mathrm{c}k}$	Chap. 7: Coefficients of the head to volume polynomial.
τ	Barrier parameter in an interior point.

Sets

$\mathcal{H}$	Set of hydro units ($\mathcal{H} \subset \Omega$).
$\mathcal{H}_k$	Set of reservoirs upstream of reservoir k.
$\mathcal{I}$	Set of successive intervals employed to define hydro generation constraints ($\iota \in \mathcal{I}$).
L^i	List of LMCs used in interval i.
$\mathcal{M}_j$	Intervals in which unit j is not available because of fixed maintenance.
N	Ranges of intervals, from i_a to i_z, where each unit $j \in \Phi$ can start its maintenance (in the B&B algorithm).
$\mathcal{U}^i$	Set of available thermal units in interval i.
η	Chap. 8: Set of units with the maintenance already assigned ($\eta \subseteq \Phi$).
Ξ	Set of units belonging to the SGC.
Φ	Chap. 2 and 8: Set of generation units whose maintenance schedule is to be optimized. Chap. 7: Set of units to be committed.
ϕ^i	Set of units already ordered.
ω	Subset of units of Ω ($\omega \subset \Omega$).
Ω	Set of generation units of the pool.
Ω_o	Ordered set with the indices of the units (loading order).
Ω_{o_j}	Ordered set with the indices of the units ordered before unit j.
$\mathbb{R}$	The set of real numbers.
$\mathbb{B}$	The set $\{0,1\}$.

Functions

$f(z)$	Density function of the demand.
$\mathcal{F}_d(i)$	Distribution of the d variable over the intervals.
$\mathcal{F}_d(i)$	Distribution of the s variable over the intervals.
m^i	Forecasted market price in interval (hour) i.
$S_\omega(z)$	Load-survival function ($S(z) = prob(\text{demand} > z)$) of still-unsupplied load after loading units in ω.
$W(x_j^i)$	Expected profit of unit j in interval i.
ψ	Quadratic objective function.

Variables

d_j^i	Binary variable that indicates whether unit j initiates maintenance in interval i.
g	Chap. 5 and 6: Vector of dual variables (multipliers of the primal constraints).
g_j^i	Chap. 7: Power generation of thermal unit j over interval i.
h_k^i	Average of the hydropower generation of reservoir k in interval i.
$\widehat{m}^i$	Market-price estimate in (hourly) interval i.
o_{RoP}	Amount of emission allowances bought or sold by the RoP.
o_{SGC}	Amount of emission allowances bought or sold by the SGC.
p_k^i	Spillage in reservoir k during interval i.
s_j^i	Binary variable that indicates the state (maintenance/in-service) of unit j in interval i.
u_j^i	Binary variable that indicates whether unit j starts up in interval i.
v_k^i	Volume of water in reservoir k at the end of interval i.
$\widehat{v}_k^i$	Approximated volume of water in reservoir k in interval i.
$w(\omega)$	Expected unsupplied energy after loading units in ω.
x_j^i	Expected energy generated by unit j during interval i.
$x_{j\|\omega}$	Expected generation of unit j ordered after units in ω.
y_k^i	Chap. 7: Volume of water discharged of reservoir k in interval i.
y	Chap. 5 and 6: Vector of primal slack variables.
z	Chap. 5 and 6: Vector of dual slack variables.
z^i	Chap. 7: Amount of zero-priced energy in (hourly) interval i.
$\widehat{z}^i$	Chap. 7: Estimate of the amount of zero-priced energy in (hourly) interval i.
α	Step length for a given direction.
Δ	Search direction.
Θ	Positive-semidefinite diagonal matrix used to regularize the factorization of the augmented-like form of Newton system.
λ_k^i	Chap. 3: Contribution of the vertex k of interval i to the current point.
π	Lagrange multiplier of the constraints of the LTGP problem transformation.
σ_k	Chap. 3: Surplus of the non-LMCs inequalities of the LTGP problem transformation.
$\widetilde{\sigma}_k^i$	Chap. 7: Equivalent water head.
ν	Chap. 3: Lagrange multiplier of the nonbasic surpluses of the LTGP problem transformation.

Bibliography

[1] A. Altman and J. Gondzio. Regularized symmetric indefinite systems in interior point methods for linear and quadratic optimization. *Optimization Methods and Software*, (11-12):275 – 302, 1999.

[2] 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.

[3] J. Barquín, E. Centeno, and J. Reneses. Stochastic market equilibrium model for generation planning. *Probability in the Engineering and Informational Sciences*, 19:533 – 546, 2005.

[4] C. Batlle and J. Barquín. A strategic production costing model for electricity market price analysis. *IEEE Transactions on Power Systems*, 20(1):67 – 74, February 2005.

[5] H. Y. Benson and D. F. Shanno. An exact primal-dual penalty method approach to warmstarting interior-point methods for linear programming. Technical report, 2005. Working paper available at `http://rutcor.rutgers.edu/~shanno/warmLP.pdf`.

[6] H. Y. Benson and D. F. Shanno. Interior-point methods for nonconvex nonlinear programming: regularization and warmstarts. Technical report, 2005. Working paper available at `http://rutcor.rutgers.edu/~shanno/l1.pdf`.

[7] 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.

[8] R.R. Booth. Power system simulation model based on probability analysis. *IEEE Transactions on Power Apparatus and Systems*, PAS-91:62 – 69, January/February 1972.

[9] J. Cabero, A. Baíllo, S. Cerisola, M. Ventosa, A. García-Alcalde, F. Perán, and G. Relaño. Medium-term integrated risk management model for a hydrothermal generation company. *IEEE Transactions on Power Systems*, 20(3):1379 – 1388, 2005.

[10] A. Conejo, R. García-Bertrand, and M. Díaz-Salazar. Generation maintenance scheduling in restructured power systems. *IEEE Transactions on Power Systems*, 20(2):984 – 992, May 2005.

[11] A. J. Conejo, F. J. Nogales, and J. M. Arroyo. Price-taker bidding strategy under price uncertainty. *IEEE Transactions on Power Systems*, 17(4):1081–1088, 2002.

[12] A. J. Conejo, F. J. Nogales, J. M. Arroyo, and R. García-Bertrand. Risk-constrained self-scheduling of a thermal power producer. *IEEE Transactions on Power Systems*, 19(3):1569–1574, 2004.

[13] A. J. Conejo and F. J. Prieto. Mathematical programming and electricity markets. *Sociedad de Estadística e Investigación Operativa, Top*, 9(1):1 – 54, 2001.

[14] 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.

[15] J. Contreras, M. Klusch, and J. B. Krawczyk. Numerical solutions to Nash-Cournot equilibria in coupled constraint electricity markets. *IEEE Transactions on Power Systems*, 19(1):195 – 206, February 2004.

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

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

[18] C. J. Day, B. F. Hobbs, and J. Pang. Oligopolistic competition in power networks: a conjectured supply function approach. *IEEE Transactions on Power Systems*, 17(3):597 – 607, August 2002.

[19] European Climate Exchange. Historical data - ECX CFI futures contract. `http://www.europeanclimateexchange.com/` (accessed July 2006).

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

[21] A. Forsgren. On warm starts for interior methods. Technical Report TRITA-MAT-2005-OS4, Dept. of Mathematics, Royal Institute of Technology, Sweden, 2005. Available at `http://www.math.kth.se/~andersf/doc/warm.pdf`.

[22] R. Fourer, D. M. Gay, and B. W. Kernigham. *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press/Brooks/Cole Publishing Company, 2002.

[23] J. Gardner, W. Hobbs, F. N. Lee, E. Leslie, D. Streiffert, and D. Todd. Summary of the panel session "Coordination between short-term operation scheduling and annual resource allocations". *IEEE Transactions on Power Systems*, 10(4):1879 –1889, November 1995.

[24] P. Gill, W. Murray, and M. Saunders. *User's Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming*, 2006. Available on-line: `http://www.sbsi-sol-optimize.com/manuals/SNOPT%20Manual.pdf`.

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

[26] P. E. Gill, W. Murray, and M. H. Wright. *Numerical Linear Algebra and Optimization*, volume 1. Addison Wesley, 1991.

[27] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins, 2nd edition, 1989.

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

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

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

[31] J. A. Gonzalez. Probabilistic production costing modeled with AMPL. *IEEE Transactions on Power Systems*, 17(2):277 – 282, May 2002.

[32] A. Pagès homepage. `http://www-eio.upc.es/~apages`.

[33] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373 – 395, 1984.

[34] R. Kelman, L. A. N. Barroso, and M. V. F. Pereira. Market power assessment and mitigation in hydrothermal systems. *IEEE Transactions on Power Systems*, 16(3):354 – 359, August 2001.

[35] B. Krasenbrink, B. Pribićević, and H.-J. Haubrich. Integrated planning of power generation and trading in a competitive market. *IEEE Power Engineering Society Summer Meeting*, 3:1271–1277, 2002.

[36] Y. Liu and F.F. Wu. Generator bidding in oligopolistic electricity markets using optimal control: fundamentals and application. *IEEE Transactions on Power Systems*, 21(3):1050 –1061, August 2006.

[37] M.K.C. Marwali and S.M. Shahidehpour. Long-term transmission and generation maintenance scheduling with network, fuel and emission constraints. *IEEE Transactions on Power Systems*, 14(3):1160 – 1165, August 1999.

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

[39] B. Mo, A. Gjelsvik, and A. Grundt. Integrated risk management of hydro power scheduling and contract management. *IEEE Transactions on Power Systems*, 16(2):216 –221, May 2001.

[40] B. A. Murtagh and M. A. Saunders. Large-scale linearly constrained optimization. *Mathematical Programming*, 14(1):41 – 72, Dec 1978.

[41] B. A. Murtagh and M. A. Saunders. *Minos 5.5 User's guide*. Technical Report SOL 83-20R, Department of Operations Research, Stanford University, Revised July 1998.

[42] N. Nabona. On the loading order in the matching of an electric load-duration curve by generation units. Technical Report DR 2006/03, Dept. Estadística i Inv. Operativa, Univ. Politèc. de Catalunya, Barcelona, 2006. Available at `http://www-eio.upc.es/~gnom`.

[43] N. Nabona and M.A. Díez. Long-term thermal multi-interval thermal generation scheduling using Bloom and Gallant's linear model. Technical Report DR9803, Dept. Estadística i Inv. Operativa, Univ. Politèc. de Catalunya, Barcelona, 1998.

[44] 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):69–772, 2001.

[45] 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. Available at `http://www-eio.upc.es/~gnom`.

[46] 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.

[47] N. Nabona and A. Pagès. A three-stage short-term electric power planning procedure for a generation company in a liberalized market. *International Journal of Electrical Power & Energy Systems*, 2006 (accepted for publication).

[48] N. Nabona and F. Rossell. Formulation of the short–term hydrothermal coordination of electricity generation and solutions through mixed integer programming. Technical Report DR1999/06, Dept. Estadística i Inv. Operativa, Univ. Politèc. de Catalunya, Barcelona, 1999.

[49] GNOM Group on Numerical Optimization Modeling. `http://www-eio.upc.es/~gnom/`.

[50] 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. Available at `http://www-eio.upc.es/~gnom`.

[51] 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*, 2005 (accepted for publication). Available online from 15th May 2006 at `http://www.sciencedirect.com/`.

[52] A. Pagès, J. Gondzio, and N. Nabona. Warmstarting for interior-point methods applied to the long-term power planning. Technical Report DR2006/04, Dept. Estadística i Inv. Operativa, Univ. Politèc. de Catalunya, Barcelona, 2006. Available at `http://www-eio.upc.es/~gnom`.

[53] 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.

[54] J. Reneses, E. Centeno, and J. Barquín. Coordination between medium-term generation planning and short-term operation in electricity markets. *IEEE Transactions on Power Systems*, 21(1):43 – 52, February 2006.

[55] T. Bjørkvoll, S. Fleten, M. Nowak, A. Tomasgard, and S. Wallace. Power generation planning and risk management in a liberalised market. *IEEE Porto Power Tech Conference*, 1, 2001.

[56] R.E. Rosenthal. A nonlinear network flow algorithm for maximization of benefits in a hydroelectric power system. *Operations Research*, 29(4):763–784, 1981.

[57] A. Rudkevich, M. Duckworth, and R. Rosen. Modeling electricity pricing in a deregulated generation industry. The potential for oligopoly pricing in a poolco. *Energy Journal*, 19(3):19 – 48, 1998.

[58] M. Shahidehpour, H. Yamin, and Z. Li. *Market operations in electric power systems*. Wiley-Interscience, 2002.

[59] J. R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. Technical Report PA 15213, School of Computer Science, Carnegie Mellon University, 1994.

[60] G. B. Shrestha, B. K. Pokharel, T. T. Lie, and S.-E. Fleten. Medium term power planning with bilateral contracts. *IEEE Transactions on Power Systems*, 20(2):627 –633, May 2005.

[61] European Comission. EU action against climate change. EU emissions trading – an open scheme promoting global innovation. `http://europa.eu.int/comm/environment/climat/pdf/emission_trading2_en.pdf` (accessed June 2006), 2005.

[62] S. Uryasev and R. Y. Rubinstein. On relaxation algorithms in computation of noncooperative equilibria. *IEEE Trans. Automat. Contr.*, 39:1263 – 1267, June 1994.

[63] J. Valenzuela and M. Mazumdar. A probability model for the electricity price duration curve under an oligopoly market. *IEEE Transactions on Power Systems*, 20(3):1250 – 1256, August 2005.

[64] R. Vanderbei. Symmetric quasi-definite matrices. *SIAM J. Optim.*, 5:100 – 113, 1995.

[65] R. J. Vanderbei. *Linear Programming: Foundations and Extensions*, volume 37 of *International Series in Operations Research & Management Science*. Kluwer, 2nd edition, 2001.

[66] M. Ventosa, A. Baíllo, A. Ramos, and M. Rivier. Electricity market modeling trends. *Energy policy*, 33:897 – 913, 2005.

[67] X.F. Wang, J.R. McDonald, and W. Xifan. *Modern Power System Planning.* McGraw Hill, 1994.

[68] A. J. Wood and B. F. Wollenberg. *Power generation operation & control.* Wiley, 1984.

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

[70] 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.