Université catholique de Louvain

École polytechnique de Louvain (EPL)

Models and Algorithms for Pricing Electricity in Unit Commitment

Dissertation presented by
Nicolas Stevens

for obtaining the Master's degree in
Mathematical Engineering

Supervisor(s)
Anthony Papavasiliou , Gauthier de Maere d'Aertrycke

Reader(s)
Yurii Nesterov, Yves Smeers

Academic year 2015-2016

Acknowledgements

I would like first to express my grate gratitude to my supervisors Anthony Papavasiliou as well as Gauthier de Maere d'Aertrycke, for supporting me and guiding me along this thesis, but also for their wise advices and help during the elaboration of the codes and models used in the work.

I am also grateful to Gauthier and the whole team at ENGIE for providing me the data without which it would have been impossible to make consistent simulations.

I would like to thank particularly Yurii Nesterov for its wise advices on the algorithmic questions which leads to the development of the efficient scheme presented in this work.

Besides, I thank Yves Smeers for agreeing to be part of the thesis jury as a reader.

As this work flags the end point of all the academic years at university, I would like to thank family and friends for having being part of these five incredible years. Finally, and not the least, I thank my better half for her support and her love.

Contents

Introduction

Energy is one of the greatest challenge in current times. It raises political concerns as well as technical and economical issues, the three being deeply linked. To give a picture of what we are talking about, in 2014 the total Belgian consumption was up to 77.1 [TWh] (including 70.47 TWh produced in Belgium) at an average price of 40.8 [€/MWh][1]. This is a huge amont of money. Therefore, the price of this commodity is a critical question. In addition to that, electricty has one of the most volatile price compared to other commodities, [17].

At the same time, the growing share of renewable energy in the market has come up with new way to think the energy and is even changing the landscape of our countries. It also adds complexity in the system and rises new challenges. Especially, the electricity market is becoming more and more uncertain. Indeed, at each moment, the electricity demand has to be fulfilled by the wind, hydro, gas, nuclear plants... However, this demand involves uncertainty and at the same time the renewable production also implies uncertainty linked to the weather forecast. In order to cope with such unpredictability, the system needs more flexibility. This flexibility among other can be provided by plants which can quickly switch on/off in case of unexpected peaks of demand or drop of renewable production. Those plants are facing substantial start-up costs or no-load costs which appears in the market auction price through their bids. Therefore, a great concern is to build a pricing model which takes into account the fixed start-up costs for those plants.

However, as it is detailed later on, the well-known marginal pricing is unable to deal with such fixed cost... Recent work, i.e. [5], has come up with a new promising pricing scheme which is about to be implemented in some U.S. states.

This work has three goals :

- to briefly expose the economical and mathematical model related to this price;
- to build an efficient algorithm able to compute the price for real applications;
- to test it on the CWE (center-west European) market in order to figure out how such pricing scheme behaves and to see if this fundamental model can predict the real price of the market.

This work is mostly composed by convex programming concepts in a background of electricity market. The first chapter is focussed on economical concepts for pricing the electricity; the second chapter develops the mathematical models; the third chapter presents an efficient algorithm and the fourth chapter displays the results of such pricing method applied to the European market.

It should be noted that the three first chapters have been built to be understood separately. The greatest contribution to the literature is probably the algorithmic scheme presented at chapter 3 (more specifically figure 3.9 which demonstrates the efficiency of the algorithm) as well as the

[1] From "Evolution des marchees de l'electricite et du gaz naturel en Belgique, Annee 2014, Communique de presse", *CREG*

application of such pricing model on the CWE market at chapter 4 (more specifically figure 4.6 which demonstrates the high quality of the pricing model).

Nomenclature

Sets	
$g \in G$	Set of generators
$t \in \mathcal{T}$	Set of hourly period horizon (let's define $T = card(\mathcal{T})$)
$s \in S_g$	Start-up type of generator $g \in G$ (e.g. *hot*, *warm*, *cold*)
Parameters	
D_t	Demand level at time $t \in \mathcal{T}$
R_t	Reserve target at time $t \in \mathcal{T}$
C_g^{NL}	No-load cost of generator $g \in G$
C_g^P	Production cost (variable cost) of generator $g \in G$
$C_{g,s}^{SU}$	Start-up cost of type $s \in S_g$ of generator $g \in G$
C_g^{SD}	Shut-down cost of generator of generator $g \in G$
$P_g^{\min}$	Minimal power level of generator $g \in G$
$P_g^{\max}$	Maximal power level of generator $g \in G$
RU_g	Ramp-up limit of generator $g \in G$
RD_g	Ramp-down limit of generator $g \in G$
SU_g	Start-up capacity of generator $g \in G$
SD_g	Shut-down capacity of generator $g \in G$
TU_g	Minimal up time of generator $g \in G$
TD_g	Minimal down time of generator $g \in G$
$T_{g,s}^{SU}$	Time interval of generator $g \in G$ for which the start-up is of type $s \in S_g$ (i.e. if a start-up occurs in t such that the previous shut-down was in $t' \in [t - T_{g,s}^{SU}; t - T_{g,s+1}^{SU}]$ then the start-up is of type s)
Variables	
$\widehat{p}_{g,t}$	Total power provided by generator $g \in G$ at time $t \in \mathcal{T}$
$p_{g,t}$	Power above $P_g^{\min}$ provided by generator $g \in G$ at time $t \in \mathcal{T}$ (i.e. the total power is given by $u_{g,t}P_g^{\min} + p_{g,t}$)
$u_{g,t}$	Commitment decision (on or off) of generator $g \in G$ at time $t \in \mathcal{T}$
$v_{g,t}$	Start-up decision of generator $g \in G$ at time $t \in \mathcal{T}$
$w_{g,t}$	Shut-down decision of generator $g \in G$ at time $t \in \mathcal{T}$
$r_{g,t}$	Ramp-up spinning reserve provided by generator $g \in G$ at time $t \in \mathcal{T}$
$\delta_{g,s,t}$	Start-up type selector, for a start-up of type $s \in S_g$ by generator $g \in G$ at time $t \in \mathcal{T}$

Chapter 1

The convex hull pricing

In the introduction, the problematic of an efficient pricing of electricity has been risen. This chapter aims to state mathematically what is behind the economical intuition in order to build a price model. For the notations we refer the reader to the nomenclature that has been detailed previously.

1.1 Economic dispatch and marginal pricing

The electricity market is known to be a complex market as it is composed by different level (day-ahead market, intra-day market...) and as it is highly coupled with physical constraints (line limits, network constraints, i.e. Kirschkoff laws, ramp limits...). As this is not a central question in the elaboration of a fundamental pricing model, let's get rid of such complexity for now (much details about physical constraints are explained at chapter 2 and on the market itself at chapter 4).

In essence, what the market does is to fulfil the electricity demand coming from the industry or from private consumers. In order to do that, many electric plants are available to provide a given amount of power at a certain cost and the market operator is responsible for dispatching the electricity among these producers. Of course, it would be wise to pick a solution which minimizes the cost of the system. Furthermore, as the demand changes constantly, the dispatch process has to be repeated at a certain frequency. In this work, the case of a **day-ahead** market cleared **hourly** is considered.

The simpler mathematical formulation of what has been described is

$$\begin{aligned}
\min_{\widehat{p}} \quad & \sum_t \sum_g C_g^P \widehat{p}_{g,t} \\
s.t.\ (\rho_t)\ & D_t = \sum_g \widehat{p}_{g,t} && \forall t \\
(\mu_{g,t})\ & \widehat{p}_{g,t} \leq P_g^{\max} && \forall g, t \\
& \widehat{p}_{g,t} \geq 0 && \forall g, t
\end{aligned} \tag{1.1}$$

which is called the **economic dispatch model**, as it only cares about dispatching the power $\widehat{p}_{g,t}$. It is obvious that this problem is **convex** as it is linear. The demand constraint is the only constraint that links all the generators together and is called the "market clearing constraint".

A crucial question when designing a market is the price at which the commodity (the electrical energy in our case) is sold. The electricity market is cleared using a **uniform pricing**. As it is

Units	U1	U2	U3
$P^{\max}$	50	40	10
C^P	10	20	30

Table 1.1

not the purpose of this work to discuss such economical model, we consider this as a fact without discussing it but in order to convince that such pricing is fair and consistent let's notice that considering a price-taker market, uniform pricing provides incentives to market participants to bid truthfully.

In model (1.1), using duality theory, the dual variable ρ_t represents the variation of the total cost of the system facing a perturbation of the demand. A useful mathematical tool in order to analyse dual variables is the KKT, which for model (1.1) can be written

$$\begin{aligned} D_t = \sum_g \widehat{p}_{g,t} & & \forall t \\ P_g^{\max} \geq \widehat{p}_{g,t} \;\perp\; \mu_{g,t} \geq 0 & & \forall g,t \\ 0 \leq \widehat{p}_{g,t} \;\perp\; C_g^P - \rho_t + \mu_{g,t} \geq 0 & & \forall g,t \end{aligned} \tag{1.2}$$

The interpretation of (1.2) is

- assuming that $P_g^{\max} > 0$, if plant g produces 0 at time t, then $\mu_{g,t} = 0$ and $\rho_t \leq C_g^P$;
- if $0 < \widehat{p}_{g,t} < P_{g,t}^{\max}$ then $\mu_{g,t} = 0$ and $\rho_t = C_g^P$;
- if $\widehat{p}_{g,t} = P_g^{\max}$ then $\mu_{g,t} \geq 0$ and $\rho_t = C_g^P + \mu_{g,t} \geq C_g^P$.

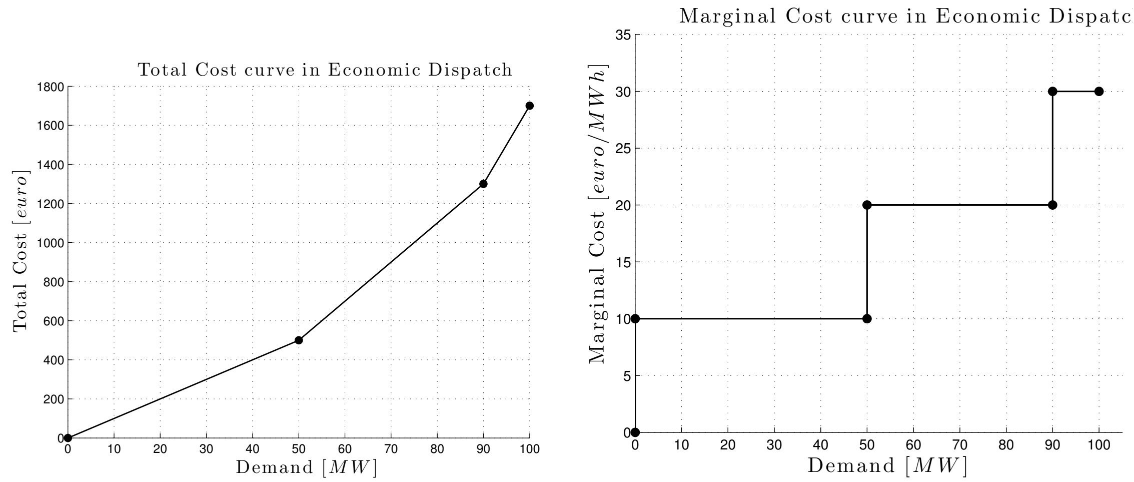
Economically this is the **marginal pricing**, i.e. the production cost of the plant which is at the margin of production (the marginal cost of the system) is chosen as the price. In convex model such as (1.1), such pricing model is an equilibrium price (this is detailed at section 1.3).

Example 1. *Let's imagine a situation, where three units are available, with the features of table 1.1. It is clear that unit U1 is the cheapest one, unit U2 the second cheapest and unit U3 the most expensive. Therefore, a optimal dispatch, i.e. solving* (1.1)*, would lead to the total cost curve of figure 1.1a : the demand is fulfilled using in priority order U1, U2 and U3 respectively. Applying the "marginal pricing", the price is the marginal cost of the system as presented at figure 1.1b.*

An important observation from figures 1.1 is that the marginal cost is increasing with the load. This sounds obvious as the problem is convex but it is important to keep it in mind. Indeed, economically, it means that high prices are indicators of scarcity. So the price is consistent with the needs of the system.

1.2 Unit commitment and marginal pricing

The problem with the simple economic dispatch (1.1), is that it only considers production costs C_g^P, without taking into account fixed costs such as start-up costs C_g^{SU} or no-load costs C_g^{NL}.



(a) (b)

Figure 1.1: Total and marginal cost curves. Based on the market described by tables 1.1 and using model (1.1).

Let's go back to the previous point. It has been highlighted that the demand changes constantly and that the production has to be adapted to the current load. In practise, this variability is often located in a group of plants (e.g. CCGT units) which provides this flexibility in the system. So start-up costs are for those plants a non-negligible part of their endured costs as they often need to switch on/off and it should be part of the model. Let's noted that the growth of renewable energy adds uncertainty and volatility in the system, making these flexible units more and more useful.

The following mathematical model extends (1.1) by including what has been discussed :

$$\begin{aligned}
&\min_{\widehat{p},u} \sum_t \sum_g C_g^P \widehat{p}_{g,t} + \sum_t \sum_g C_g^{SU} v_{g,t} \\
&s.t. D_t = \sum_g \widehat{p}_{g,t} && \forall t \\
&\quad \widehat{p}_{g,t} \leq P_g^{\max} u_{g,t} && \forall g,t \\
&\quad \widehat{p}_{g,t} \geq P_g^{\min} u_{g,t} && \forall g,t \\
&\quad v_{g,t} - w_{g,t} = u_{g,t} - u_{g,t-1} && \forall g,t \\
&\quad \widehat{p}_{g,t} \geq 0; \quad u_{g,t} \in \{0,1\} && \forall g,t
\end{aligned} \tag{1.3}$$

This model is called the **unit commitment**, as it includes the on/off status $u_{g,t}$ of the plants (and start-up/shut-down status $v_{g,t}$ and $w_{g,t}$). It is obvious that this problem is **non-convex** as it is an integer program.

This is a bad news as it means that strong duality does not hold and the pricing rules has to be built from scratch.

A pricing model mixing commitment and marginal pricing has been introduced by O'Neill and is briefly presented in [5]. It consists of three steps :

- solve the integer model (1.3);
- fix the commitment variables to their optimum level $u_{g,t} = u^*_{g,t}$;
- re-solve the model (1.3) the commitments being fixed (convex problem) and use the dual variables associated with the market clearing constraint as the price such as section 1.1.

Let's have a look at the performances of such pricing scheme on a simple example.

Example 2. *Let's design a small example, very similar from example 1 but involving start-up costs and minimum power. The data are given at table 1.2. Let's solve the program* (1.3) *using data of table 1.2. The total cost of the system depending on the demand is given at figure 1.2a.*

It is clear from the data that unit U1 is the cheapest one whatever the demand. Therefore, as long as the demand is below 50 [MW], U1 is providing the whole load. As soon as the demand exceeds 50 [MW], as the start-up of U2 combine with its production cost is more expensive than using U3[1]*, U3 fulfils the additional load between 50 and 60 [MW], i.e. the slope of the total cost is 30. As soon as the load exceeds 60 [MW], U2 has to be committed. As its minimal output power is 20 [MW], the production of U3 drops to 0 and the one of U1 to 40 [MW]. Therefore, producing one additional MW cost 10 [€/MWh], i.e. the slope is worth 10, and the total cost is* $100 + 20 \times 20 + 10 \times 40 = 900$ *(so there is a discontinuity). When the demand exceeds 70 [MW], as U2 is already on, producing an additional MW is cheaper with U2 than U3 so U2 produces the marginal load and the slope is 20. Finally, when the demand goes up 90 [MW], U3 is assigned to this additional load.*

The "marginal cost" of the system (i.e. the cost for producing one more MW) depending on the demand is given at figure 1.2b.

Pick the price as this marginal cost is exactly the O'Neill idea : the units are committed and as the commitment is fixed, the price is defined as the marginal cost.

Let's point out two major observations :

- *Unlike example 1, the marginal cost is non-increasing. This is a real concern, as a high price as it is the case for a load of 55 [MW] at figure 1.2b does not indicate scarcity at all. Therefore, the economical reasons sustaining such price are rather fuzzy and would not be understand by the market participants.*
- *Let's consider a demand of 65 [MW]. Considering a marginal pricing scheme, the price would be 10 [€/MWh] (see figure 1.2b). As a reminder the production of units 1, 2 and 3, as asked by the market operator for such price are* $\widehat{p}_{U1} = 45$, $\widehat{p}_{U2} = 20$ *and* $\widehat{p}_{U3} = 0$. *Therefore, the profits of each units are* $\pi_{U1} = 10 \times 45 - 10 \times 45 = 0$€, $\pi_{U2} = 10 \times 20 - 20 \times 20 - 100 = -300$€*and* $\pi_{U3} = 0$€. *So U2 is facing losses! Indeed, for such price U2 would wish to produce 0. So this price does not provide the right incentives to the market participants to produce the required power.*

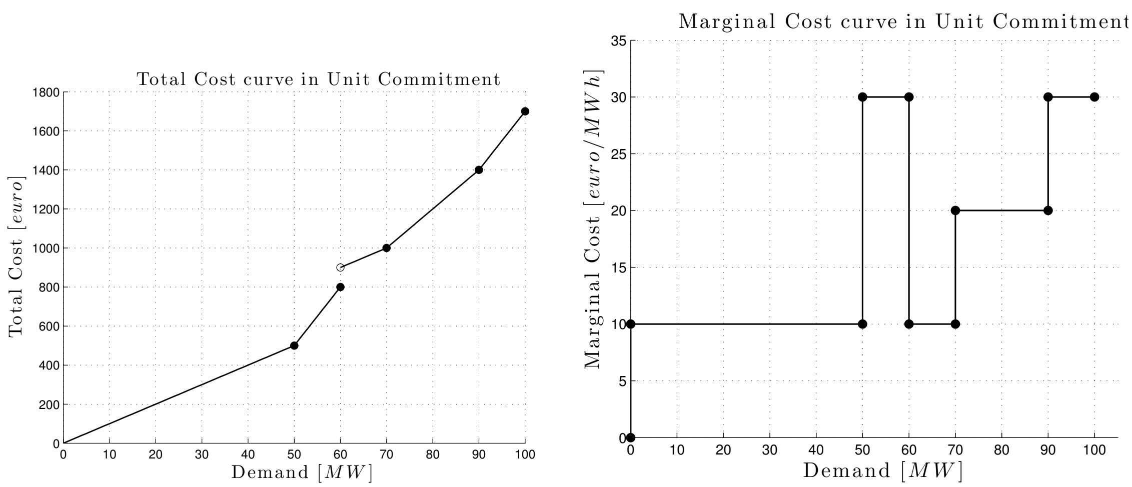
Another model, could concist of relaxing integrality of the binary variables and relplace $u \in \{0, 1\}$ by $0 \leq u \leq 1$. But this also leads to poor results, [5].

This lack of a consistent model appeals for a more efficient pricing scheme.

[1] Let p be the power for which U3 is more expensive than U2 : $20p + 100 < 30p$, for $p > 10$, so U2 is more profitable as soon as there is more than 10 MW to produce. Below a demand of 60 MW, as U1 manages 50 MW there are less then 10 MW to produce by another unit and therefore U3 is used.

Units	U1	U2	U3
$P^{\min}$	0	20	0
$P^{\max}$	50	40	10
C^{P}	10	20	30
C^{SU}	0	100	0
initial status	off	off	off

Table 1.2



(a) (b)

Figure 1.2: Total and marginal cost curves. Based on the market described by tables 1.2 and using model (1.3).

1.3 Equilibrium and uplifts paiements

Before presenting the pricing model for which the rest of the materials is dedicated, let's make a brief digression on the notion of equilibrium.

Definition 1. *A price ρ is called an equilibrium if given such price, the producers have the incentives to commit and to produce such that the market clears.*

Or equivalently, the producers maximize their own utility and the market clears. The very appyling property of an equilibrium is that an equilibrium if it exists, is efficient ("first welfare theorem", [17]).

A competitive equilibrium exists *if and only if* strong duality holds (e.g. convex problems such as (1.1)). This theorem is admitted without proof. However the left sense implication is obvious from the reasonning of section 1.1 using the KKT. A formal proof involves to express the decentralized problems of (1.1) and is presented at Theorem 9 in [17].

This is a bad new as it leads to the following statement

Corollary 1. *The non-convex problem* (1.3) *does not always admit a competitive equilibrium price.*

This makes the goals and the hope for positive results of this work rather low...

However, let's imagine how a market could be built using model (1.3). Whatever the price model be, the market has to be cleared. Furthermore, the previous corollary implies that finding a price such that the producers maximize their utility and the market clears is an unreachable goal. This brings the need of *side-payments*, also called *uplifts payments.* Given the market clearing price ρ, the producers costs and the quantity each producer is asked to provide, the producers may incur : losses (i.e. $\rho\widehat{p}_g - C_g^P\widehat{p}_g < 0$) or more generally opportunity costs (i.e. a market participant with a positive profit but which does not maximize his utility). The side payments are exactly introduced to compensate such losses or opportunity costs.

Definition 2. *Let $\pi_g^{as-bid} = \sum_t \rho_t\widehat{p}^*_{g,t} - f_g(\widehat{p}^*_{g,t})$ be the optimum profit of plant g facing price ρ (i.e.* ***maximum as-bid profit****), where $\widehat{p}^*_{g,t}$ is the optimal response of g to price ρ and $f_g(\widehat{p}^*_{g,t})$ is the corresponding total cost. Let $\pi_g^{as-cleared} = \sum_t \rho_t\widehat{p}^f_{g,t} - f_g(\widehat{p}^f_{g,t})$ be the profit of plant g under a forced power $\widehat{p}^f_{g,t}$ and a price ρ (i.e.* ***maximum as-cleared quantity profit****); then*

$$uplift_g = \pi_g^{as-bid}(\rho) - \pi_g^{as-cleared}(\rho)$$

The introduction of such uplift payments allows us to define more reasonable goals (using the same terminology as [13]) :

Definition 3. *The combination of market price revenues and uplift payments are* ***incentive compatible*** *if the market participants do not have incentives to deviate from their cleared quantities.*

Remark 1 Let's notice that whereas competitive equilibrium price is purely market-based incentive, the introduction of uplift payments kills this property. Therefore, it should be noted that uplift payments are NOT wished but are a necessary drawback in order to design a pricing scheme on model (1.3). Indeed, side payments are discriminative and not transparent.

1.4 Unit commitment and convex hull pricing

The last section has given some hope for finding a pricing model using uplift payments which is *incentive compatible*. The most appropriate candidate is denoted as the *Convex Hull Pricing*. This idea has been introduced in 2007 by W. Hogan [5] to tackle the problem of pricing non-convex unit commitment in the specific case of electricity market.

There are many ways to understand the origin and the very natural aspect of such pricing model. Using what has been presented so far, three intuitive ways of understanding the convex hull price are successively presented : mathematically, graphically and economically.

1.4.1 Mathematically

As explained at section 1.2, for the non-convex unit commitment (1.3), strong duality does not hold. However, it is known that in convex situation such as economic dispatch (1.1), the dual variable associated with the market clearing constraint provides an efficient price. Therefore, a natural transposition of this idea to non-convex problem (1.3) is to apply a **Lagrangian relaxation** of the market clearing constraint, introducing Lagrangian multipliers ρ_t and maximizing the obtained relaxed program according to ρ (weak duality) :

$$\max_{\rho} \left\{ \begin{array}{ll} \displaystyle\min_{\widehat{p},u} \sum_t \sum_g C_g^P \widehat{p}_{g,t} + \sum_t \sum_g C_g^{SU} v_{g,t} + \sum_t \rho_t \left(D_t - \sum_g \widehat{p}_{g,t} \right) & \\ s.t.\ \widehat{p}_{g,t} \leq P_g^{\max} u_{g,t} & \forall g,t \\ \quad\ \ \widehat{p}_{g,t} \geq P_g^{\min} u_{g,t} & \forall g,t \\ \quad\ \ v_{g,t} - w_{g,t} = u_{g,t} - u_{g,t-1} & \forall g,t \\ \quad\ \ \widehat{p}_{g,t} \geq 0; \quad u_{g,t} \in \{0,1\} & \forall g,t \end{array} \right\} \tag{1.4}$$

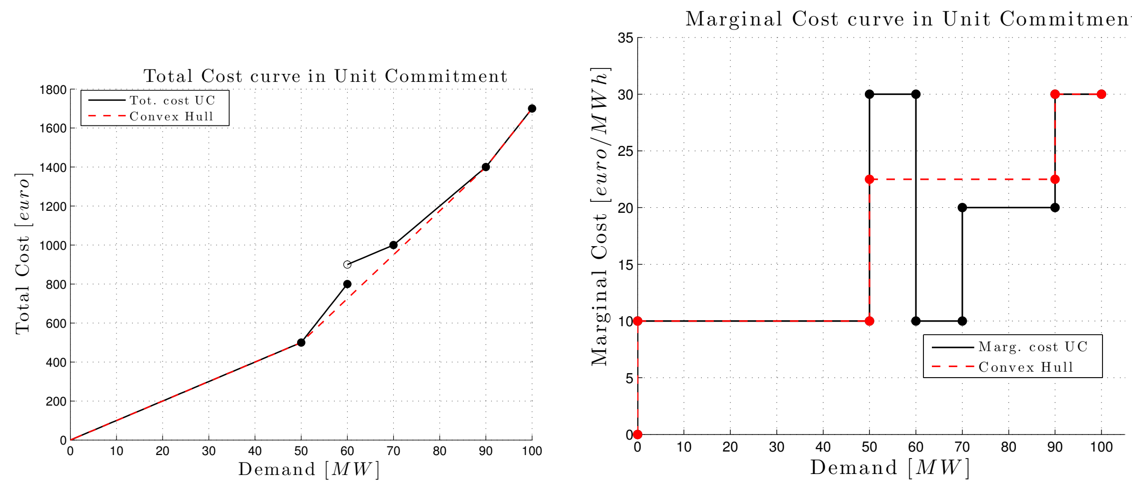
In this Lagrangian problem, ρ_t is the convex hull price (CHP). This is the practical computational way to find these CHP. It should be notice that solving (1.4) is a computational challenge (taking into account that the problem has to be solved on a hourly time frame, so the price should be available in a very few minutes...). Chapter 3 addresses the algorithmic questions and more specifically, example 7 illustrates graphically the behaviour of an algorithm solving problem (1.4) using data of example 2.

The link between (1.4) and the graphical or economical interpretation is pointed out in the next sections.

1.4.2 Graphically

Example 2 highlighted that a non-increasing price is not wished and therefore O'Neill pricing seems inappropriate. The next example addresses this problem.

Example 3. *Let's consider the data of example 2. Graphically, the most straightforward way to deal with the non-increasing issues of the marginal cost is to build the* ***convex hull*** *of the total cost curve. This has been done at figure 1.3a. The slope of this convex hull is represented at figure 1.3b and is of course increasing with the load. Taking this slope as the price provides successively 10, 22.5 and 30. Let's again highlight two major observations.*



(a) (b)

Figure 1.3: Total and marginal cost curves. Based on the market described by tables 1.2 and using model (1.3) and the convex hull model.

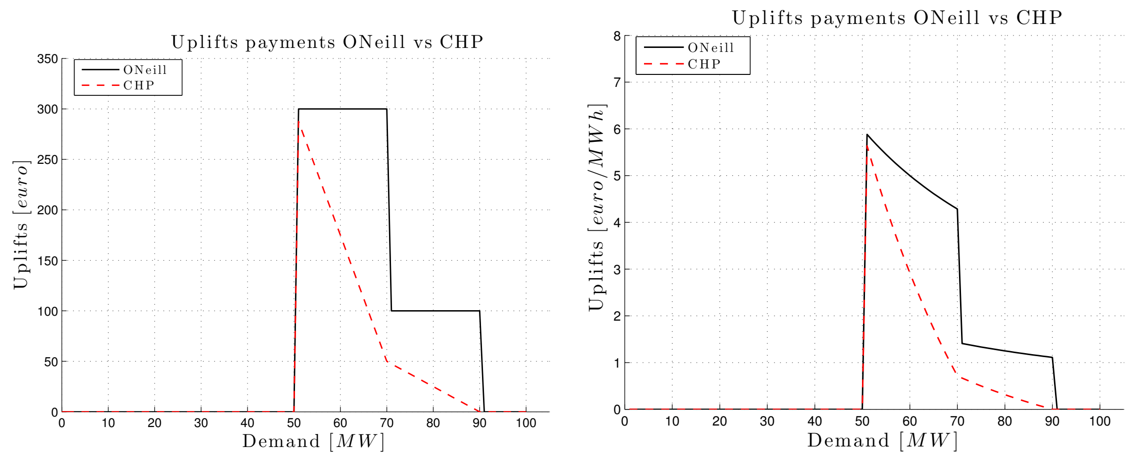
- *Let's consider a demand of 65 [MW]. The CHP for such demand is 22.5 [€/MWh]. For such price, both U1 and U2 would like to produce at their maximum level. However, for U2 :* $22.5 - 20 = 2.5$*; whereas for U1:* $22.5 - 10 = 12.5$*. So the uplifts needed by U1 for each MW below its maximum level are higher than those of U2. Therefore the production of U1 will be preferred by a system operator applying the "min uplift" rule. For a demand of 65 [MW], U1 is asked to produce 45 [MW] and U2 20 [MW]. Their profit for such price is* $\pi_{U1} = (22.5 - 10) \times 45 = 562.5$€*and* $\pi_{U2} = 20 \times (22.5 - 20) - 100 = -50$€*. So U2 makes a few losses. For such price, U1 as well as U2 would wish to produce 50 MW instead of the 45 MW and 40 MW instead of 20 asked by the operator, so uplift payments are required (i.e.* $5 \times (22.5 - 10) + 20 \times (22.5 - 20) = 112.5$ *[€]) but it is quite obvious that this situation is not as dramatic as the one of example 2.*

- *In a sense, CHP is what is the most close to a marginal pricing and preserves the convex properties such as an increasing price with the load.*

This graphical interpretation suggests the name of "convex hull" pricing. It should be noted that even if the graphical interpretation is elegant, it does not provide a computable way to come up with the convex hull price. Indeed, computing a convex hull in higher dimension is a hopeless goal.

1.4.3 Economically

Section 1.3 pointed out that in non-convex situations, uplifts are necessary in order to achieve an incentive compatible state. It also highlighted that uplift are not wished. From such analysis, a natural efficient price would consist of finding ρ_t which minimizes the uplifts such as defined at



(a) (b)

Figure 1.4: Total uplifts payments (left figure) and uplift payment per MW (right figure). Based on the market described by tables 1.2 and using model (1.3) and the convex hull model.

definition 2, i.e.

$$\min_{\rho} \ \pi_g^{as-bid}(\rho) - \pi_g^{as-cleared}(\rho) \tag{1.5}$$

It appears that program (1.4) and (1.5) are equivalent. This is proved formally in the next section but in order to convince that it is intuitively the case, let's test it on a small example.

Example 4. *Let's use the data of example 2 and 3. Computing the uplifts such as suggested by program 1.5 for the O'Neill model and the convex hull price model (equivalently program* (1.4) *or using figures 1.3), gives the results of figure 1.4. Clearly the uplifts required by the convex hull pricing model are substantially lower than those of O'Neill.*

Remark Regarding what has been described at section 1.3, it should be notice that the convex hull price is not expected to be an equilibrium price. But it is a good candidate for an efficient pricing scheme due to some of its desirable properties which are detailed in the next section.

1.4.4 Convex hull pricing main properties

Let's show that the three previous intuitions are strictly equivalent.

Theorem 1. *Convex hull price minimizes the uplifts payments as defined at definition 2.*

The following proof is similar to the one presented in [5] or [13].

Proof. Let's first reformulate problem (1.4) :

$$\max_{\rho} \left\{ \min_{\widehat{p},u\in X} \sum_g C_g^P \widehat{p}_g + \sum_g C_g^{SU} v_g + \rho \left(D - \sum_g \widehat{p}_g \right) \right\} \tag{1.6}$$

$$= \max_{\rho} \left\{ \rho D + \min_{\widehat{p},u\in X} \left\{ \sum_g C_g^P \widehat{p}_g + \sum_g C_g^{SU} v_g - \rho \sum_g \widehat{p}_g \right\} \right\} \tag{1.7}$$

$$= \max_{\rho} \left\{ \rho D - \max_{\widehat{p},u\in X} \left\{ \rho \sum_g \widehat{p}_g - \sum_g f(p_g) \right\} \right\} \tag{1.8}$$

where f_g denote the cost function of g. It is possible to figure out that uplifts are deeply linked with the notion of duality gap. Indeed, as the primal model (1.3) is non-convex, there might be a duality gap with the dual problem (1.4) (which is a lower bound of (1.3)). Denote $v(p^*)$ the optimal solution of (1.3). Then using (1.8) :

$$\begin{aligned}
\text{duality gap } &= v(p^*) - \max_{\rho} \left\{ \rho D - \max_{\widehat{p},u\in X} \left\{ \rho \sum_g \widehat{p}_g - \sum_g f(p_g) \right\} \right\} \\
&= v(p^*) + \min_{\rho} \left\{ -\rho D + \max_{\widehat{p},u\in X} \left\{ \rho \sum_g \widehat{p}_g - \sum_g f(p_g) \right\} \right\} \\
&= \min_{\rho} \left\{ \max_{\widehat{p},u\in X} \left\{ \rho \sum_g \widehat{p}_g - \sum_g f(p_g) \right\} - (\rho D - v(p^*)) \right\} \\
&= \min_{\rho} \left\{ \max_{\widehat{p},u\in X} \left\{ \rho \sum_g \widehat{p}_g - \sum_g f(p_g) \right\} - (\rho \sum_g p_g^* - v(p^*)) \right\} \\
&= \min_{\rho} \left\{ \pi_g^{as-bid}(\rho) - \pi_g^{as-cleared}(\rho) \right\} \quad (\geq 0)
\end{aligned}$$

as $D = \sum_g \widehat{p}_g^*$. Indeed, $\max_{\widehat{p},u\in X} \left\{ \rho \sum_g \widehat{p}_g - \sum_g f(p_g) \right\}$ is the profit maximization program of the generators facing the price ρ so the "pay as-bid profit"; whereas $(\rho \sum_g p_g^* - v(p^*))$ is the profit facing price ρ if the units are forced to provide $\widehat{p}_g^*$ so it is the "pay as-cleared profit". □

This fundamental theorem makes the link between the economical intuition and the formal mathematical model (1.4) obvious.

Corollary 2. *Convex hull price applied to a convex problem such as the economic dispatch* (1.1)*, provides the competitive equilibrium price, i.e. the "marginal" price.*

Proof. The proof is straightforward applying strong duality to problem (1.4). □

This fact is crucial as it involves that such pricing scheme is consistent with marginal pricing in convex situations.

Theorem 2. *Problem* (1.4) *may admit several optimum solutions, i.e. the convex hull price is not always unique.*

Hour	1	2	3	4	5	6	Lagrangian value
Demand [MW]	30	40	80	80	80	10	
Price 1 [€/MWh]	10	10	20	22.5	20	10	4175
Price 2 [€/MWh]	10	10	22.5	20	20	10	4175
Price 3 [€/MWh]	10	10	20	20	22.5	10	4175
Price 4 [€/MWh]	10	10	20.25	20.25	22	10	4175
Price 5 [€/MWh]	10	10	20.83333	20.83333	20.83333	10	4175

Table 1.3: Results of the convex hull pricing on a six periods study case of table 1.2

This is an important fact and example 5 illustrates it.

Example 5. *Let's use again the study case presented at table 1.2. Unlike the previous examples, let's now have a look at a multiple hours horizon considering a six hours problem with the demand of table 1.3.*

Intuitively, the price at period 1,2 and 6 should be 10 [€/MWh] as the only required unit is U1. For periods 3,4 and 5 it is less straightforward. It sounds logical that the price would be located between 20 and 30 [€/MWh]. It is also expected that the price endures a "peak" corresponding to the start-up cost of U2. However, if in a one period study case, CHP may be computed easily using for instance the graphical interpretation of example 3, it is clearly not as simple in 6 dimensions. Therefore, let's assume that we have an algorithm providing the CHP (see chapter 3).

This algorithm provides the solution called "price 1" at table 1.3 and the corresponding Lagrangian optimum value. The prices of 20 and 22.5 sound familiar from example 3. However, it would have been intuitively more convincing to have the "price spike" of 22.5 at the time U2 has to start, i.e. in period 3 such as price 2. Indeed, evaluating the Lagrangian function at price 2 of table 1.3 comes up with the same Lagrangian value as price 1, i.e. price 2 is also an optimum, meaning a valid convex hull price. The same hold for price 3 of table 1.3.

Furthermore, as problem (1.4) *is concave (this is proved at chapter 3), any convex combination of these three prices is also an optimum (see for instance prices 4 or 5 at table 1.3).*

Let's point out two observations :

- *On the one hand, this sounds logical as for prices 1, 2 and 3, units U1, U2 as well as U3 receive the same overall payment.*
- *On the other hand, if price 2 may sound natural (a spike of price at the time the unit have to start-up), the others are less intuitive in the sense that they can not be understood looking at a single period without considering the whole horizon. It arises a general conclusion on the convex hull pricing :* ***it is not a well-known economic concept*** *(e.g. unlike marginal pricing) which might be a concern for the market participants as the price does not always seem economically intuitive.*

1.5 Reserve

As the system may incur unexpected failures such as generator outages or transmission line collapses, it would be interesting to build a mechanism dealing with such contingencies. A natural approach consists of asking the generators to keep a spare capacity which can quickly turns out into dispatch power in case a critic scenario occurs. This stand-by power is called *reserve*. Lots of current markets trade both energy and reserve.

It can be mathematically modelled as a positive variable $r_{g,t}$ which does not incur costs as it is not a produced power, but shares capacity constraints and ramp constraints with energy... Typically, system operators require a certain amount R_t of available reserve. Including reserve in unit commitment model (1.3) leads to[2]

$$\begin{aligned} \min_{\widehat{p},u} \; & \sum_t \sum_g f_g(p_{g,t}, v_{g,t}) & \\ s.t. D_t = & \sum_g \widehat{p}_{g,t} & \forall t \\ & \sum_g r_{g,t} \geq R_t & \forall t \\ & \widehat{p}_{g,t}, r_{g,t}, u_{g,t}, v_{g,t}, w_{g,t} \in X^g & \forall g, t \\ & \widehat{p}_{g,t}, r_{g,t} \geq 0; \quad u_{g,t} \in \{0,1\} & \forall g, t \end{aligned} \tag{1.9}$$

where f_g are the non-convex total costs and the technical constraints of each generator are aggregated in X_g as they are discussed in chapter 2.

Program (1.3) has one coupling constraint, the demand constraint, and one convex hull price for the single commodity $\widehat{p}_{g,t}$ associated with this constraint. Program (1.9) has two coupling constraints, i.e. energy demand and reserve requirement, and two commodities associated, i.e. power and reserve, for which there should be a convex hull price.

Having said that, Lagrangian relaxation (1.4) can be extended to

$$\max_{\rho^{elec},\rho^{res}} \left\{ \begin{array}{ll} \min\limits_{\widehat{p},u} \; \sum\limits_t \sum\limits_g f_g(\widehat{p}_{g,t}, v_{g,t}) + \sum\limits_t \rho_t^{elec} \left(D_t - \sum\limits_g \widehat{p}_{g,t} \right) + \sum\limits_t \rho_t^{res} \left(R_t - \sum\limits_g r_{g,t} \right) & \\ s.t. \;\; \widehat{p}_{g,t}, r_{g,t}, u_{g,t}, v_{g,t}, w_{g,t} \in X^g & \forall g, t \\ \quad\;\; \widehat{p}_{g,t}, r_{g,t} \geq 0; \quad u_{g,t} \in \{0,1\} & \forall g, t \\ \quad\;\; \rho_t^{res} \geq 0 & \forall t \end{array} \right\}. \tag{1.10}$$

In this program, ρ^{elec} denotes the convex hull price for energy (as defined in the previous section) and ρ^{res} denotes the convex hull price for reserve. As this work is focussed on energy prices, we are more interested in the influence of reserve on the energy prices ρ^{elec} than in reserve price ρ^{res} itself[3].

Reserve is disputed briefly as this thesis goal is not to tackle this issue in detail despite it is a broad and tricky question. Therefore, only *ramp-up spinning reserve* (provided by on-line generators) is considered. Some tests performed at chapter 4 discuss the obtained price depending on including reserve or not.

[2]A design issue is whether there should be an equality or inequality in reserve requirement constraint... Here inequality constraint has been chosen which leads to non-negative reserve prices in the dual problem.

[3]For information, the remuneration of reserve is generally *pay-as-bid* rather than uniform pricing

Chapter 2

A tight and compact formulation of the Unit Commitment

This chapter introduces the comprehensive model of Unit Commitment (UC). The UC problem has already been introduced in chapter 1. In this chapter, the technical and physical constraints are specified, i.e. model (1.3) is extended to a more suitable model for real applications. The nomenclature of this chapter is broad but has been clearly established at the beginning of the dissertation. The main constraints are detailed with the required quotations from the literature. Furthermore, this chapter is focussing on explaining the main compromises encountered while expressing a MIP model as well as issues of the tightness and compactness . For the sake of clearness, some features are ignored (pump-storage plants, network constraints, fuel index...). A comprehensive model including each of these features is detailed at appendix A.

2.1 Tightness and compactness for MIP programs

When dealing with modelling problems, there often are several ways to express a given constraint. Some appears to be more effective when solving it using a solver such as `CPLEX`. A natural and intuitive explanation of distinct computational behaviour between models expressing the same physical constraint is the *compactness*. The compactness is basically the size of the problem, i.e. the number of constraints, variables and non-zero elements in the constraints matrix (this parameter can have dramatic impacts on the computational performances, [7]). A formulation of the problem with a biggest amount of variables, constraints and non-zero elements is naturally expected to be slower.

Another parameter which influences the computational performances is the *tightness*. This is less intuitive and requires to detail the main philosophy of the algorithmic scheme used by `CPLEX`. Let's suppose that we are solving a minimisation MIP program. The main body of the algorithm used by `CPLEX` is *branch and bound* (*bnb*) algorithms speeded up by some heuristics. A *bnb* requires to find at each step (each node of the tree) a lower bound (from a linear relaxation) and an upper bound (a feasible integer solution). The quality of those bounds impacts the computational performances as it changes the number of nodes to be checked, [7]. A problem is said to be tight if the solution of the LP relaxation is closed to the integer solution. Or similarly, a problem is tight if the size of the search space is reduced. Figure 2.1 illustrates graphically the tightness of a formulation on a 2-D example. Both blue and red formulations have the same integer domain but the blue formulation

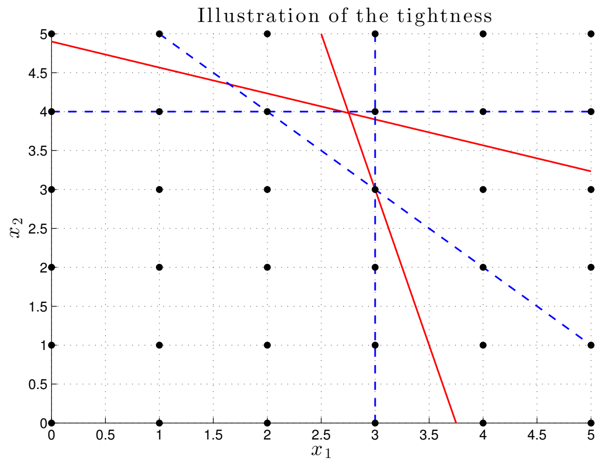


Figure 2.1: Two ways of defining the same integer domain : one defined by the under part of the red constraints and another defined by the under part the blue ones). The blue is actually the convex hull formulation which is the tightest formulation.

will be solved in one *bnb* iteration as each vertice of the polygonal domain is integer (i.e. it is a convex hull) whereas the red requires more iterates to cut through the domain.

The compactness and the tightness are generally opposite targets. Indeed tightening a model generally consists of adding constraints or variables damaging the compactness and reciprocally. Expressing a problem as an efficient deal between tightness and compactness is a modelling challenge.

2.2 The model

This section presents the model defining the unit commitment problem. The next points introduce step by step the constraints specifying the model. A discussion of the tightness and compactness is provided as well as references. The section is concluded by the final model.

2.2.1 1-*bin* or 3-*bin* model

The first step in the process of building a unit commitment model is the definition of the variables. In our case, the binary status of each plant can be modelled by a single binary variable $u_{g,t}$ (1-*bin* model). Indeed it would be possible to entirely build a model using a single set of binary variables. However, it is a common knowledge in unit commitment formulation that explicitly defining start-up binary variables $v_{g,t}$ and shut-down binary variables $w_{g,t}$ (3-*bin* model) is computationally more efficient, [15]. In fact, almost each and every papers quoted in this chapter use this 3-*bin* model; so it is the case in the rest of the material. Variables $v_{g,t}$ and $w_{g,t}$ are maked meaningful through the following equation :

$$u_{g,t} - u_{g,t-1} = v_{g,t} - w_{g,t} \qquad \forall g, t \qquad (2.1)$$

This constraint could be called "logical constraint" as it links variables $u_{g,t}$, $v_{g,t}$ and $w_{g,t}$ [1].

2.2.2 Generation capacity limits

These constraints force the produced power to lie in the interval $[P^{min}; P^{\max}]$. A trivial way of expressing these constraints is

$$\begin{aligned} \widehat{p}_{g,t} + r_{g,t} &\leq P_g^{\max} u_{g,t} \\ \widehat{p}_{g,t} &\geq P_g^{\min} u_{g,t} \end{aligned}$$

where $\widehat{p}_{g,t}$ is the power output. [7], [4] and [6] show that a tighter and equally compact formulation could be

$$\begin{aligned} p_{g,t} + r_{g,t} &\leq (P_g^{\max} - P_g^{\min}) u_{g,t} \\ p_{g,t} &\geq 0 \end{aligned}$$

where $p_{g,t}$ is the power above the minimal level (i.e. the total power is given by $u_{g,t} P_g^{\min} + p_{g,t}$).

Using this last formulation and including the start-up and shut-down power (SU_g and SD_g, being respectively the power available when starting up and the power level required in order to shut down) gives :

$$\begin{aligned} p_{g,t} + r_{g,t} &\leq (P_g^{\max} - P_g^{\min}) u_{g,t} - (P_g^{\max} - SU_g) v_{g,t} - (P_g^{\max} - SD_g) w_{g,t+1} && \textbf{if } TU_g \geq 2,\ \forall g, t \\ p_{g,t} + r_{g,t} &\leq (P_g^{\max} - P_g^{\min}) u_{g,t} - (P_g^{\max} - SU_g) v_{g,t} - \max(SU_g - SD_g; 0) w_{g,t+1} && \textbf{if } TU_g = 1,\ \forall g, t \\ p_{g,t} + r_{g,t} &\leq (P_g^{\max} - P_g^{\min}) u_{g,t} - (P_g^{\max} - SD_g) w_{g,t+1} - \max(SD_g - SU_g; 0) v_{g,t} && \textbf{if } TU_g = 1,\ \forall g, t \end{aligned}$$

where two cases are considered, $TU_g = 1$ or $TU_g \geq 0$, in order to avoid cases where $v_{g,t} = 1$ and $w_{g,t+1} = 1$ (see [7], [4] for more details).

2.2.3 Minimum up/down time & variable start-up cost

A minimum up/down time constraint which is tight and compact is presented in [7], [4] and [6] as well as in [11] :

$$\begin{aligned} \sum_{i=t-TU_g+1}^{t} v_{g,i} &\leq u_{g,t} && \forall g, t \in [TU_g, T] \\ \sum_{i=t-TD_g+1}^{t} w_{g,i} &\leq 1 - u_{g,t} && \forall g, t \in [TD_g, T]. \end{aligned}$$

The first one is meant to be interpreted as : if generator g has been started up during the last TU_g periods (i.e. $\sum_{i=t-TU_g+1}^{t} v_{g,i} = 1$) then g should be "on" on t (i.e. $u_{g,t} = 1$); or if $u_{g,t} = 0$

[1] Let's notice in [7] that even if $v_{g,t}$ and $w_{g,t}$ are not required to be explicitly defined as binary variables because of the "logical constraint" which forces them to be binary whatever they would have been declared as continuous; empirical results show in [7] that `CPLEX` can take benefit of this binary information even if it is not a mathematical necessity. So it is wise to declare explicitly these variables as binary in the code.

then it can not have been started during the last TU_g periods, i.e. $\sum_{i=t-TU_g+1}^{t} v_{g,i} = 0$. The same reasoning can be applied to the second constraint.

It is proven in [11] that these constraints combine with (2.1) define facets of the convex hull of the up/down time inequalities. It is also proven in [4] and [6] that these constraints combined with the former tight generation capacity constraints form a convex hull (i.e. the tightest possible formulation). This makes the problem as easy to solve as a linear program (see results in those papers). Of course, adding additional constraints could break this statement but still those constraints remain tight and well formulated.

Let's complexify the model by introducing variable start-up cost. This variable start-up cost describes several type of start-up (e.g. *cold*, *warm*, *hot*) according to the former shut-down (i.e. which determines the temperature of the generator). It can be expressed as in [7] :

$$\delta_{g,s,t} \leq \sum_{i=T^{SU}_{g,s}}^{T^{SU}_{g,s+1}-1} w_{g,t-i} \qquad \forall g, s \in S_g, t \in [T^{SU}_{g,s+1}, T]$$

$$\sum_{s \in S_g} \delta_{g,s,t} = v_{g,t} \qquad \forall g, t$$

where the start-up cost is $\sum_{s \in S_g} C^{SU}_{g,s} \delta_{g,s,t}$. The binary variable $\delta_{g,s,t}$ is used to select the start up type $s \in S_g$. The second constraint imposes that if there is a start-up in t then it should be of a certain type s. The first constraint forces $\delta_{g,s,t}$ to be equal to 0 for the s type which does not incurred a shut down on its given interval $[t - T^{SU}_{g,s}; t - T^{SU}_{g,s+1}]$. As the start-up cost is increasing according to the length of the off period, the program will naturally choose the very last shut-down. This formulation is compact and effective but it is not a facet of the convex hull.

A convex hull formulation of the variable start-up cost combined with the minimum up/down time has been proposed in [10]. This formulation uses graph theory representation. Each node corresponds to a time period. The aim is to find a path from node 0 to node T using arcs either $\phi_{i,j}$ which represents an off period starting in i and finishing in j or $\psi_{i,j}$ which represents an on period starting in i and finishing in j. More specifically, let's define an arc $(i, j) \in A_1$ as a switch-off in i followed by a start-up in j and an arc $(i, j) \in A_2$ as a start-up in i followed by a switch-off in j. The minimum up and down time are included as bounds in the sets A_1 and A_2 : $\{(i, j) \in A_1 : TD_g \leq j - i\} \subseteq A_1$ and $\{(i, j) \in A_2 : TU_g \leq j - i\} \subseteq A_2$. The model is expressed as :

$$\sum_{(i,t) \in A_1} \phi^g_{i,t} - \sum_{(t,j) \in A_2} \psi^g_{t,j} = 0 \qquad \forall g, t$$

$$\sum_{(i,t) \in A_2} \psi^g_{i,t} - \sum_{(t,j) \in A_1} \phi^g_{t,j} = 0 \qquad \forall g, t$$

$$\sum_{i \leq t; j > t} \psi^g_{i,j} = u_{g,t} \qquad \forall g, t$$

$$\sum_{t-T^{SU}_{g,s} \leq i < t-T^{SU}_{g,s+1}} \phi^g_{i,t} = \delta_{g,s,t} \qquad \forall g, t, s \in S_g$$

where the first equation forces an off period finishing in t to be followed by an on period starting in t. The second equation expresses the contrary. The third equation forces $u_{g,t}$ to be 1 if t lies inside an on period. The last equation links the start-up selector $\delta_{g,s,t}$ to the correct start up type s according to the time when the former off period has started (i.e. if there is an off period finishing in t, this is if $\exists i,$ such as $\phi_{i,t} = 1$ and if the i such as $\phi_{i,t} = 1$ lies in $[t - T^{SU}_{g,s}; t - T^{SU}_{g,s+1}]$ then the start-up in t is of type s).

This formulation is more tight than the previous one as it is proved in [10] that it is a facet of the convex hull. As far as the compactness is concerned, $T \times T$ new variables $\phi_{i,j}$ and $\psi_{i,j}$ have been introduced for the same amount of constraints.

For the sake of a comprehensive literature review, lets' finally notice that some papers go even further in the moddeling accuracy by introducing temperature and heat trasfert variables to capture the start-up profile and cost more precisely ([15]).

2.2.4 Ramp up/down limits

A natural way of defining the ramp-up/down constraints is [9]

$$\begin{aligned} \widehat{p}_{g,t} - \widehat{p}_{g,t-1} &\leq RU_g + M v_{g,t} && \forall g, t \\ \widehat{p}_{g,t-1} - \widehat{p}_{g,t} &\leq RD_g + M w_{g,t} && \forall g, t \end{aligned}$$

where a *big-M* parameter has been introduced to make the constraints consistent with the start-up and shut-down. But this is also damaging dramatically the tightness of these expressions.

Let's now define $p_{g,t}$ as the power output above the minimal level $P_g^{\min}$ (see generation capacity limits section) and let's include the reserve. Then the ramp constraints can be written as [7] :

$$\begin{aligned} (p_{g,t} + r_{g,t}) - p_{g,t-1} &\leq RU_g && \forall g, t \\ -p_{g,t} + p_{g,t-1} &\leq RD_g && \forall g, t \end{aligned}$$

which does no more require a *big-M* parameter. So this formulation is tighter than the previous one.

[3] proposes tighter formulations of the ramps constraints. First, a two hours dependant ramp constraint is introduced. An adapted version of these, compatible with our $p_{g,t}$ as defined in the nomenclature, can be expressed as :

$$\begin{aligned} (p_{g,t+1} + r_{g,t+1}) - p_{g,t} &\leq (P_g^{\min} + RU_g) u_{g,t+1} + (SU_g - 2P_g^{\min} - RU_g) v_{g,t+1} - P_g^{\min} u_{g,t} + P_g^{\min} w_{g,t+1} && \forall g, t \\ p_{g,t} - p_{g,t+1} &\leq (SD_g - 2P_g^{\min} - RD_g) w_{g,t+1} + (P_g^{\min} + RD_g) u_{g,t} - P_g^{\min} u_{g,t+1} + P_g^{\min} v_{g,t+1} && \forall g, t \end{aligned}$$

The apparent complexity is just introduced in order to make these constraints tight when the units start-up or shut-down. This formulation is tighter than the previous one and equally compact.

Then [3] also presents multi hours ramp constraints, combined with generation capacity constraints. The philosophy of such constraints is that if a generator is at a certain level p_0 at time t, it would never reach a level above $p_0 + kRU_g$ at time $t + k$. These constraints are proved to shape a convex hull. It is the tightest possible formulation, nevertheless it is less compact than the two hours ramp constraints.

As we are working with hourly period horizons, most of the ramp constraints are expected to be non-biding. Therefore, we restrict ourself to the two hours ramp constraints.

Features / Papers	[7]	[4]	[6]	[11]	[3]	[10]	[15]
Logical constraint (3 bin-variables model)[a]	√	√√	√√	√√	√√		√
Generation limits	√√ [b]	√√	√√		√√		√
Ramp up/down	√		√		√		√[d]
Multiple periods ramp-up/down					√√		
Minimum up/down time	√√ [c]	√√	√√	√√	√√ [c]	√√	√√[d] [c]
Start-up cost	√		√		√	√	√
Variable start-up cost	√					√√	√
Start-up profile			√√				
Ancillary services	√		√ [d]				
Start-up/Shut-down capacity	√	√	√		√		√[d]
Network constraints			√ [d]				

[a]This feature is specified here as some common models in the literature use 1 binary variable representation.
[b]Proven in [6]
[c] Proven in [11]
[d] In the appendix of the paper

Table 2.1: Summary of the different versions of the feature which can be found in the literature. "√" indicates a feature which is detailed in the correspondent paper. "√√" points out a feature detailed in the corresponding paper and which is proven to be tight (in the paper or another). Finally, the red " √" highlights where the real innovation of the paper is located.

2.2.5 Literature review summary

Table 2.1 provides a useful tool which gives an overview of the 7 articles that were used to build this chapter. It sums up the unit commitment features available in the quoted papers.

2.2.6 The complete model

Let's finally express our complete unit commitment model. The generation capacity constraints using $p_{g,t}$ as the power above the minimal level are hold. The minimum up/down time as well as the variable start-up cost are expressed as in [7] (the formulation [10] using graph modelling is not retained). The two-period ramp formulation from [3] are hold. Finally, the complete model can be written as equations (2.2).

As introduced in the previous chapter, we are interested here in finding the convex hull price by solving the Lagrangian relaxation of (2.2), where equation (2.2b) is relaxed. This is the topic of the next chapter. The full model including network constraints, pump-storage facilities... is presented at appendix A.

$$\min \sum_g \sum_t \left[C_g^{NL} u_{g,t} + C_g^P (P_g^{\min} u_{g,t} + p_{g,t}) + \sum_{s \in S_g} C_{g,s}^{SU} \delta_{g,s,t} + C_g^{SD} w_{g,t} \right] \tag{2.2a}$$

$$\text{s.t. } D_t = \sum_{g \in G} \left[P_g^{\min} u_{g,t} + p_{g,t} \right] \qquad \forall t \tag{2.2b}$$

$$\sum_g r_{g,t} \geq R_t \qquad \forall t \tag{2.2c}$$

$$p_{g,t} + r_{g,t} \leq (P_g^{\max} - P_g^{\min}) u_{g,t} - (P_g^{\max} - SU_g) v_{g,t} - (P_g^{\max} - SD_g) w_{g,t+1} \qquad \textbf{if } TU_g \geq 2,\ \forall g,t \tag{2.2d}$$

$$p_{g,t} + r_{g,t} \leq (P_g^{\max} - P_g^{\min}) u_{g,t} - (P_g^{\max} - SU_g) v_{g,t} - \max(SU_g - SD_g; 0) w_{g,t+1} \qquad \textbf{if } TU_g = 1,\ \forall g,t \tag{2.2e}$$

$$p_{g,t} + r_{g,t} \leq (P_g^{\max} - P_g^{\min}) u_{g,t} - (P_g^{\max} - SD_g) w_{g,t+1} - \max(SD_g - SU_g; 0) v_{g,t} \qquad \textbf{if } TU_g = 1,\ \forall g,t \tag{2.2f}$$

$$u_{g,t} - u_{g,t-1} = v_{g,t} - w_{g,t} \qquad \forall g,t \tag{2.2g}$$

$$(p_{g,t+1} + r_{g,t+1}) - p_{g,t} \leq (P_g^{\min} + RU_g) u_{g,t+1} + (SU_g - 2P_g^{\min} - RU_g) v_{g,t+1} - P_g^{\min} u_{g,t} + P_g^{\min} w_{g,t+1} \qquad \forall g,t \tag{2.2h}$$

$$p_{g,t} - p_{g,t+1} \leq (SD_g - 2P_g^{\min} - RD_g) w_{g,t+1} + (P_g^{\min} + RD_g) u_{g,t} - P_g^{\min} u_{g,t+1} + P_g^{\min} v_{g,t+1} \qquad \forall g,t \tag{2.2i}$$

$$\sum_{i=t-TU_g+1}^{t} v_{g,i} \leq u_{g,t} \qquad \forall g, t \in [TU_g, T] \tag{2.2j}$$

$$\sum_{i=t-TD_g+1}^{t} w_{g,i} \leq 1 - u_{g,t} \qquad \forall g, t \in [TD_g, T] \tag{2.2k}$$

$$\delta_{g,s,t} \leq \sum_{i=T_{g,s}^{SU}}^{T_{g,s+1}^{SU}-1} w_{g,t-i} \qquad \forall g, s \in S_g, t \in [T_{g,s+1}^{SU}, T] \tag{2.2l}$$

$$\sum_{s \in S_g} \delta_{g,s,t} = v_{g,t} \qquad \forall g,t \tag{2.2m}$$

$$p_{g,t}, r_{g,t} \geq 0 \ ;\ u_{g,t}, v_{g,t}, w_{g,t}, \delta_{g,s,t} \in \{0,1\} \qquad \forall g,t,s \tag{2.2n}$$

Chapter 3

Algorithmic schemes for finding the convex hull price

The previous chapters highlighted the necessity of developing efficient tools for solving Lagrangian problems such as problem (1.4). In this chapter several possible algorithmic schemes are studied. All of them have been implemented by the author in Python using GAMS as a modelling language. The illustrations as well as most of the comments, pro's and con's of the presented schemes comes from numerical tests performed by the author.

The material of this chapter is organized as follows. Section 3.1 shows how the previous model (1.4) can be treated in a decentralized form. Section 3.2 describes the mains characteristics of the problem which can be exploited to develop a proper algorithm. Section 3.3 displays a brief literature review and analyse more deeply the more promising algorithms. Finally section 3.5 shows some results and compares the different schemes presented in 3.3.

From the tests performed at section 3.5, it is concluded that the Level method ([8]) clearly overpasses the others in terms of robustness and computational efficiency, in this specific case of computing the convex hull price.

3.1 Decentralized formulation of the problem

Let's apply the Convex hull price formulation to our unit commitment model (2.2). For the sake of a clear and compact formulation, the problem is expressed here as a single node problem without reserve requirement (but it is straightforward to include them), constraints (2.2d) to (2.2n) are aggregated in the set denoted by X_g and de cost function (2.2a) is denoted by $f_g = f_g(u_{g,t}, p_{g,t}, v_{g,t}, w_{g,t}, \delta_{g,s,t})$.

$$\min_{u,v,w,p} \sum_g \sum_t f_g \tag{3.1a}$$

$$s.t. \quad D_t = \sum_{g \in G} \left[P_g^{\min} u_{g,t} + p_{g,t} \right] \qquad \forall t \tag{3.1b}$$

$$(u_{g,t}, p_{g,t}, v_{g,t}, w_{g,t}, \delta_{g,s,t}) \in X_g \qquad \forall g, t \tag{3.1c}$$

Applying the Lagrangian relaxation of (3.1b) and looking for the optimum of the dual program provides the following system.

$$\max_{\rho} L(\rho) = \left\{ \begin{array}{ll} \min\limits_{u,v,w,p,\delta} \sum\limits_g \sum\limits_t f_g + \sum\limits_t \rho_t \left(D_t - \sum\limits_{g \in G} \left[P_g^{\min} u_{g,t} + p_{g,t} \right] \right) & \\ s.t. \quad (u_{g,t}, p_{g,t}, v_{g,t}, w_{g,t}, \delta_{g,s,t}) \in X_g & \forall g,t \end{array} \right\} \tag{3.2}$$

Furthermore, rearranging the terms using the property that $\min f = -\max -f$:

$$\max_{\rho} L(\rho) = \left\{ \sum_t \rho_t D_t - \sum_g \max_{\substack{u,v,w,p,\delta \\ \in X_g}} \left\{ \sum_t \rho_t \left[P_g^{\min} u_{g,t} + p_{g,t} \right] - f_g \right\} \right\} \tag{3.3}$$

Problem (3.3) is the problem studied in the rest of this chapter. The last manipulation has just decoupled the whole problem, replacing a big problem such as the UC by a set of smaller problems to solve, later called "profit maximisation" :

Profit maximization program

$$\max_{\substack{u,v,w,p,\delta \\ \in X_g}} \left\{ \sum_t \rho_t \left[P_g^{\min} u_{g,t} + p_{g,t} \right] - f_g \right\} \qquad \forall g \tag{3.4}$$

which can be solved independently a generator at a time. This fact bring some comments. First, let's notice that in general, solving G "little problems" requires less computational effort then solving a G times bigger problem. Second, as all those problems are independent, they can be practically solved on a computer using multi-threading leading to competitive computational time. Finally, thinking economically, it highlights the decentralized frame of the market and the individual incentives of each generator to operate given the market price ρ.

3.2 Features and topological characteristics of the problem

This section details the characteristics of function $L(\rho)$ as expressed at (3.3).

Theorem 3. *Function $L(\rho)$ is defined $\forall \rho \in \mathbb{R}^T$. Or in other words, $\forall \rho \in \mathbb{R}^T$, problem (3.4) has an admissible solution.*

Proof. This is straightforward from the fact that ρ does not appear in any constraints defined by X_g in (3.2). □

Theorem 4. *Function $L(\rho)$ is concave in ρ.*

Proof. Let $(u^*, v^*, w^*, p^*, \delta^*)$ be the optimal reactions to $\alpha\rho_1 + (1-\alpha)\rho_2$, $\alpha \in [0,1]$. Then using (3.2)

$$L(\alpha\rho_1 + (1-\alpha)\rho_2) = \sum_g \sum_t f_g^* + \sum_t (\alpha\rho_1^t + (1-\alpha)\rho_2^t) \left(D_t - \sum_{g \in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right)$$

$$
\begin{aligned}
&= \alpha \left(\sum_g \sum_t f_g^* + \sum_t \rho_1^t \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right) \right) \\
&+ (1-\alpha) \left(\sum_g \sum_t f_g^* + \sum_t \rho_2^t \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right) \right) \\
&\geq \alpha \left(\min_{\substack{u,v,w,p,\delta \\ \in X_g}} \sum_g \sum_t f_g + \sum_t \rho_1^t \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t} + p_{g,t} \right] \right) \right) \\
&+ (1-\alpha) \left(\min_{\substack{u,v,w,p,\delta \\ \in X_g}} \sum_g \sum_t f_g + \sum_t \rho_2^t \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t} + p_{g,t} \right] \right) \right) \\
&= \alpha L(\rho_1) + (1-\alpha) L(\rho_2)
\end{aligned}
$$

□

Let's notice that concavity implies continuity. As it is a concave maximisation problem, from convex theory we know that any local maxima is a global maximum. And from the theorems 3 and 4 there exists a global maximum ρ^*.

Definition 4. *$g(x)$ is a Supgradient[1] of f in x; i.e. $g(x) \in \hat{\partial} f(x)$, if and only if*

$$f(x) + \langle g(x), (\overline{x} - x) \rangle \geq f(\overline{x})$$

Theorem 5. *Let $(u^*, v^*, w^*, p^*, \delta^*)$ be the optimal reactions to ρ. Then $g = \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right)$ is a supgradient of L in ρ; i.e. $\left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right) \in \hat{\partial} L(\rho)$*

Proof. Let $\overline{\rho} \neq \rho$. Then using (3.2)

$$
\begin{aligned}
L(\overline{\rho}) &= \min_{\substack{u,v,w,p,\delta \\ \in X_g}} \sum_g \sum_t f_g + \sum_t \overline{\rho_t} \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t} + p_{g,t} \right] \right) \\
&\leq \sum_g \sum_t f_g^* + \sum_t \overline{\rho_t} \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right) \\
&= \sum_g \sum_t f_g^* + \sum_t \rho_t \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right) \\
&\qquad - \sum_t \rho_t \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right) + \sum_t \overline{\rho_t} \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right) \\
&= L(\rho) + \sum_t (\overline{\rho_t} - \rho_t) \left(D_t - \sum_{g\in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^* \right] \right)
\end{aligned}
$$

[1] We denote it *supgradient* (as it is concave optimization) to distinguish it from the *subgradient* (the corresponding concept in convex optimization).

which, by definition 4 of the supgradient, proofs the statement. □

Theorem 6. *Function $L(\rho)$ is a piecewise linear function.*

Proof. (Sketch) From theorem 5 the supgradients of L depend mainly of the commitment decisions $u_{g,t}$ of each generator at each time period. As the number of possible commitments is finite, the number of possible supgradients is finite which makes L piecewise linear. □

In fact, each face of the Lagrangian function L corresponds to a possible commitment. E.g. the Lagrangian relaxation function of a problem with 3 generators and 2 time periods has at most $2^{2\times 3} = 64$ faces. Such amount of possible commitments can grows fast when increasing the dimension of the problem. E.g. for a 20 generators problem with 24 time periods (still rather small problem!), there are at most $2^{20\times 24} = 3.1217 \times 10^{144}$ faces (the state-of-the-art super computer in 2015 do not overcome 10^{18} FLOPS : enumerating the previous example possible commitments would take him $10^{126}[s] \simeq 3.1710 \times 10^{18}[year]$...)

Example 6. *Let's have a look at a small example : a reduced version of the scarf example, [12]. Let's consider a single node case with five generators described as at table 3.2 (the missing parameters are supposed to be equal to 0 or ignored). A two hours situation is considered with a demand $D = [30; 40]$ MW.*

The Lagrangian function is given at figure 3.1 and is obviously piecewise linear.

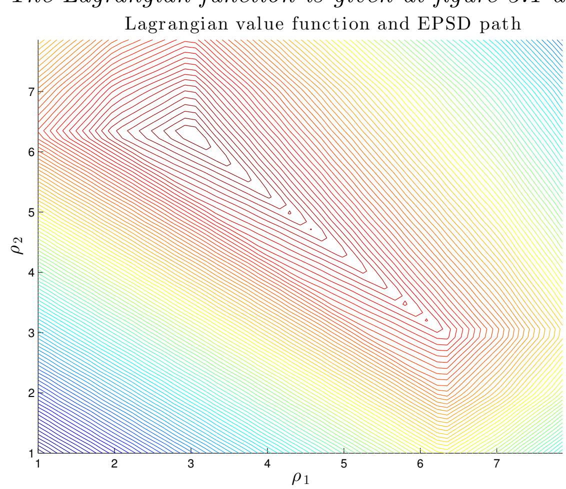


Figure 3.1: Lagrangian value function for the "scarf" example 6

G	C_g^P	C_g^{SU}	$P_g^{\min}$	$P_g^{\max}$
SMOKESTACK01	*3*	*53*	*0*	*16*
SMOKESTACK02	*3*	*53*	*0*	*16*
SMOKESTACK03	*3*	*53*	*0*	*16*
HIGH_TECH01	*2*	*30*	*0*	*7*
HIGH_TECH02	*2*	*30*	*0*	*7*
MED_TECH01	*7*	*0*	*2*	*6*

Figure 3.2: Generators data of the scarf example.

3.3 Algorithmic schemes

Considering the results from the previous section, the problem (3.3) is *concave* and *piecewise linear*. Furthermore, we are dealing with *non-smooth* optimisation (maximization) with a *first order oracle* (at each test point ρ, the value of the function as well as its supgradient is avaliable). Let's notice that the call of this oracle is expensive as it implies to solve MIP profit maximization programs (3.4) for each and every plants... Let's define Q as the domain for the price ρ (generally a bounding box which can be economically interpreted as *price caps*). The stopping criterion will be precised later, but it will be seen that for the main algorithm, upper and lower bounds UB and LB are available at each iterate. Finally, the dimension of the problem is expected to be lower than one hundred as the objective is to find the price on a horizon of one to three days (so between 24 and 72 hours).

Problem class

Model	$\max_{\rho \in Q} L(\rho)$, L concave piece-wise linear function of ρ
Oracle	First order local black box which provides a supgradient
Approximation solution	$\rho : \frac{UB(\rho)-LB(\rho)}{\lvert UB(\rho)\rvert} \leq \epsilon$
Dimension of the problem	< 100

Figure 3.3 graphically displays how any algorithmic schemes tackling this problem should work. In any case we are dealing with iterative algorithms : a sequence of iterate $\{\rho_k\}_{k=0}^{\infty}$ is generated until the the stopping criterion is met. As mention, the supgradient as well as the Lagragian value function are available by solving the "slave programs", i.e. the profit maximization programs (3.4) of each generator, given a trial price. Providing these informations the algorithm attempts to generate the iterates sequence : some schemes (1) just update the prices based on the former supgradient information (e.g. subgradient method); (2) reduce the searching domain iteratively (cutting plane methods, e.g. the ACCPM method); or (3) use some approximation model for the Lagrangian function (methods with complete data, e.g. Kelley's method). In any case a new test price is provided and sent to the slave. If the stopping criterion is met the process quits.

Some of the following quoted articles directly tackle the convex hull price problem while others ate related to general Lagrangian function optimization or even general convex non-smooth optimization.

A well known easy to implement scheme, popular for non-smooth optimization is the *subgradient scheme* ([8, 9]). This scheme neither reduces the domain nor uses approximation model function. At each trial price ρ^k, the supgradient g_k is computed and the price is updated performing a step h_k in that direction. As we are dealing with maximization program, the step is naturally made in direction g_k and not $-g_k$. In order to assure global convergence of the scheme, the update rule of h_k is expected to have the following behaviour [8] : $\sum_{k=0}^{\infty} h_k = \infty$; $h_k \to 0$ and $h_k > 0$. As it is not a strictly ascend method, an next iterate may be worst then the current one, so one have to keep track of the current best lower bound. The behaviour of this method is illustrated at figure 3.4a, where the subgradient path **oscillates** from one side of an edge to another.

One advantage of this scheme is the simplicity of implementation. The theory predict that the subgradient method converges with a **sub-linear** rate till the optimum. Experimental results show

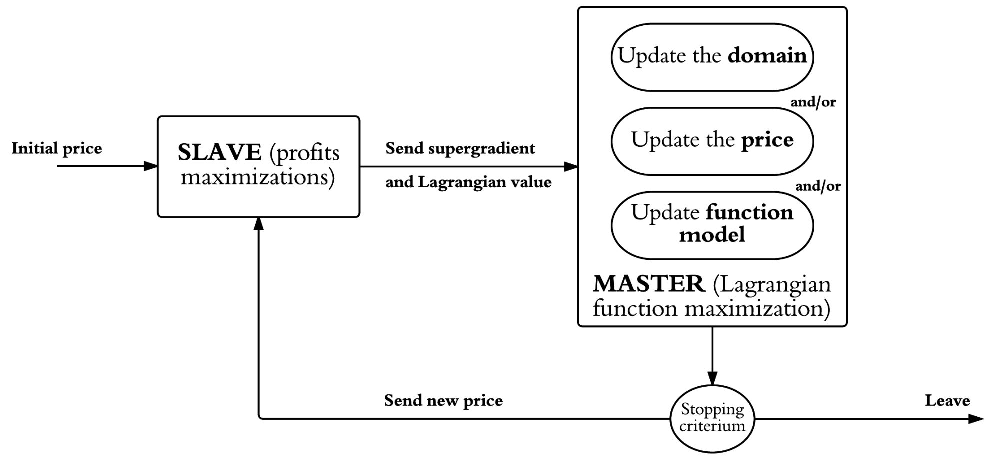


Figure 3.3: General mechanism of an iteration scheme for solving the Lagrangian maximization problem.

that in practise this rate of convergence can not be improved and is exactly consistent with what the theory predicts. Surprisingly, this scheme is proved to be the optimum method for general non-smooth optimization (not depending on the dimension of the problem), [8]. However, when looking at "low" dimension problems (<100) there might be more optimistic schemes. Finally, it should be noted that the algorithm do not provide any "upper bound" which makes the quality of the solution hard to estimate...

One may think to improve such scheme, exploiting the piecewise linear property of the Lagrangian function. In regards of figure 3.4a, one idea would be to "catch the edge", as illustrated at figure 3.4b, instead of oscillate from one side to another as the subgradient method does. This intuitive reasoning leads to the "extreme-point subdifferential" (EPSD) algorithmic scheme developped in [18, 19]. This method starts behaving such as a sugbradient scheme. When it crosses an edge, i.e. when the value of the supgradient changes, two possible cases are considered : either the iterate has just jumped on a face with a smaller supgradient, then the scheme continues as usual; or it just jumped on a face with a higher supgradient (more steep) which can be seen as a "mistake". In that case, the algorithm "catches" the edge. In order to compute the supgradient pointing in the direction of the edge, the following reasoning is triggered. At each testing point ρ, the oracle sends **each and every optimum solutions of the MIP** problems (3.4)[2]. Given the n optimum dispatches and commitments solutions[3] of each generator g for price ρ_k, let's denote them $(p_{g,n}^{k*}, u_{g,n}^{k*})$, the algorithm attempts to compute the supgradient pointing in the direction of the edge. This is done by finding the positive coefficients $\lambda_{g,n}$ such that $\sum_n \lambda_{g,n} = 1 \ \forall g \in G$ which minimize $||D - \sum_g \sum_n \lambda_{g,n} \left[u_{g,n}^{k*} P_g^{\min} + p_{g,n}^{k*}\right]||_2$ (we attempt to find the combination of the

[2] Let's go back to our previous remark : each face of the Lagrangian function corresponds to a possible commitment. If by any chance, a price ρ is located on an edge, i.e. at the intersection of two or more faces, then the commitment of each intersecting faces are optimum.

[3] `CPLEX` provides a useful tool for finding all the optimum solutions of a MIP program, called the *`CPLEX` solution pool.*

$(p_{g,n}^{k*}, u_{g,n}^{k*})$ which minimizes the violation magnitude of the relaxed constraint; i.e. $(p_{g,n}^{k*}, u_{g,n}^{k*})$ is projected on the set of (p_g, u_g) such that the demand target is met). The supgradient following the edge is then given by $g_k = D - \sum_g \sum_n \lambda_{g,n}^* \left[u_{g,n}^{k*} P_g^{\min} + p_{g,n}^{k*}\right]$.

As proven in [19], this method converges in a finite number of iterates to the optimum solution. However, the implementation of such scheme is intricate and above all requires to explore deeply the branch and bound tree at each iterate in order to find every optimum solutions of the profit maximization programs, as explained above. Furthermore some numerical tests highlights the lack of robustness of the algorithm. Finally, a comparison in [19] of EPSD vs. bundles methods highlights the promising performances of the latest which prompts us to look for alternative solutions.

Having a look back at figure 3.3, the subgradient scheme as well as EPSD update the prices directly. The next scheme is based on recursively updating the domain. In our case, as stated in theorem 3, the domain Q is bounded. At each iterate, our first order oracle provides us a supgradient which can be used to design a *cut* in the domain Q. This process iteratively shrinks the searching domain by adding cuts. This class of method are the so called "cutting plane" methods. One question remains, how should the next testing point be chosen? One popular choice is the Analytic Center Method (ACCPM, [2, 8] for the theory and quoted in [19, 16] as a reference for some comparison tests), which is studied an detailed at section 3.3.1.

Some may think to improve ACCPM by computing an approximation of the analytical center as it requires a couple of Newton iterations... [16] tackles this problem using the Subgradient Mid-Point (SMP) as an approximation of the analytical center. This is, given a bunch of cuts, a testing point x and a supgradient $\partial L(x)$, the mid-point between x and the first cut encountered in the direction of the supgradient. However, such strategy is hopeless as it leads, such as the subgradient method to oscillations (see figure 3.4c). [16] suggests some tricks to deals with that problem but they are practically hard to implement and present lacks of robustness... However, as it will be pointed out at section 3.3.1, SMP provides a good warming point for the Newton process used for ACCPM.

The subgradient scheme as well as the ACCPM are general method for non-smooth convex optimisation. None of them takes directly advantages of the piecewise linear shape of our function, as EPSD does. The next two methods [8] are based on *model* of the non-smooth function, suitable for our piecewise linear Lagrangian function, where at each iterate the model of the function is updated (see figure 3.3). Indeed, as the function is piecewise linear, each supgradient provided by the oracle, is not only an ascend direction but also provides a way of building a supporting hyperplane of the Lagrangian function. These methods are called *methods with complete data* as they keep in memory the informations about the function. These are "cutting plane methods" where the "cuts" are made over the function itself (the supporting hyperplanes) instead of in the domain. Again one question remains, how should the next testing points be chosen? Two popular methods are studied : the Kelley method [8, 9] is detailed at section 3.3.2 and the Level method [8] is detailed at section 3.3.3.

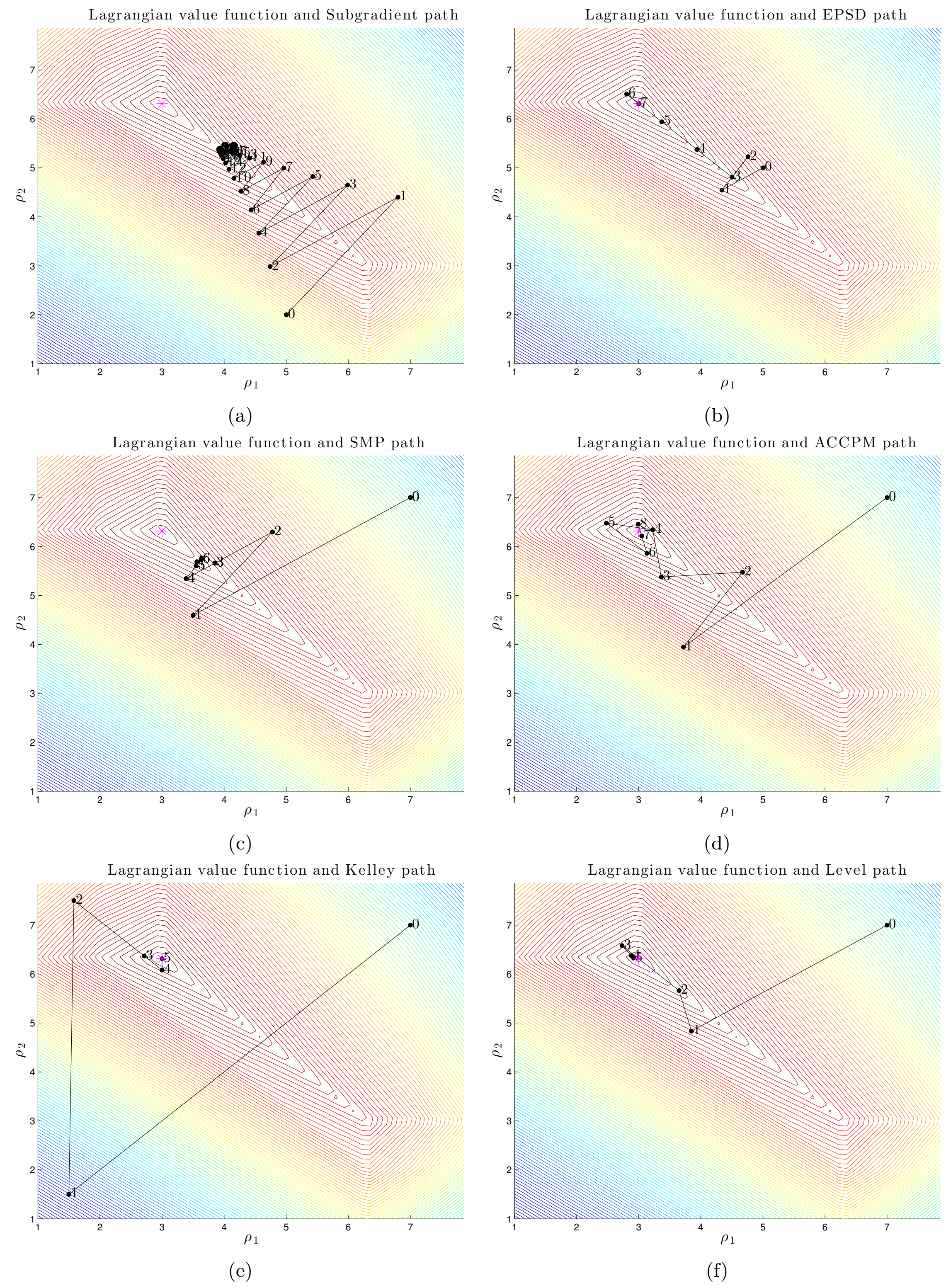


Figure 3.4: Let's solve example 6 using the developed schemes. The domain is initially bounded as $1.5 \leq \rho_1, \rho_2 \leq 7.5$.

3.3.1 Analytical center cutting plane method (ACCPM)

As stated before, this scheme aims to find and ϵ-optimum solution by iteratively shrinking the domain. This is done by defining cuts.

Theorem 7. *Let Q be the initial domain of our problem and let $\{\rho_k\}_{k=0}^{\infty}$ be a sequence in Q. Let g_k be the supgradient at iterate ρ_k such as defined at theorem 5. Then the optimal solution ρ^* of the Lagrangian maximization problem* (3.3) *satisfies*

$$\rho^* \in \{\rho \in Q | \langle g_k, \rho - \rho_k \rangle \geq 0 \ \forall k\} \tag{3.5}$$

Theorem 7 provides a way to shrink the searching domain. A question remains how to choose the sequence $\{\rho_k\}_{k=0}^{\infty}$ where the supgradient are evaluated. As each cut should discard the greatest possible volume, a natural and popular choice [2, 8] is the analytical center.

Definition 5. *Let $S_k = \{x | \langle a_i, x \rangle \leq b_i \quad \forall i = 0..k\}$ be our polygon region. Then the analytic center of S_k is defined as the optimum solution of*

$$\min_x \quad -\sum_{i=0}^{k} \log(b_i - \langle a_i, x \rangle) \tag{3.6}$$

So finding the analytical center requires to minimize an analytic barrier function using Newton's scheme. This brings a last issues : to provide to the Newton scheme a starting point x_0. In general, finding an initial point for a Newton process is a tough task, [2]. However, our particular problem allows us to find a rather good starting point, inspired from [16], called the *subgradient mid point* (SMP).

Definition 6. *Let g_k be the supgradient at iterate ρ_k. Defining $a_k = -g_k$ and $b_k = \langle -g_k, \rho_k \rangle$ then $S_k = \{x \in Q | \langle a_i, x \rangle \leq b_i \quad \forall i = 0..k\}$ is our polygon region at step k defined by the cuts. Then the Subgradient Mid Point (SMP) is defined as*

$$\rho_k^{SMP} = \rho_k + \frac{\alpha}{2} g_k \tag{3.7}$$

where $\alpha = \min_{\alpha > 0} \left\{ \frac{b_i - \langle a_i, \rho_k \rangle}{\langle a_i, g_k \rangle}, \quad i = 1..k \right\}$ i.e. this is the mid point between the current iterate and the first encountered cut facing it in the direction of the supgradient.

Now we have everything we need to define our ACCPM method given at algorithm 1.

3.3.2 Kelley's cutting plane method

This method as well as the one presented in section 3.3.3 is based on *model* of the non-smooth function.

Definition 7. *Let Q be the initial domain of our problem and let $\{\rho_k\}_{k=0}^{\infty}$ be a sequence in Q. Let g_k be the supgradient at iterate ρ_k such as defined at theorem 5. Then*

$$\hat{L}(\rho, k) = \min_{i=0..k} [\langle g_i, \rho - \rho_i \rangle + L(\rho_i)] \tag{3.8}$$

is a model for our Lagrangian function $L(\rho)$.

Algorithm 1 $ACCPM(\rho^0, \epsilon, iter^{\max})$

let $[g_i]_{i=0}^k$ be the supgradient at $[\rho^i]_{i=0}^k$; L_k be the Lagrangian function value at ρ^k;
$Lowerbound \leftarrow -\infty$
for $k \leftarrow 0, iter^{\max}$ **do**
  solve profit maximization
  $g_k \leftarrow \left(D_t - \sum_{g\in G}\left[P_g^{\min} u_{g,t}^* + p_{g,t}^*\right]\right)$
  $L_k \leftarrow \sum_t \rho_t^k D_t - \sum_t \rho_t^k \left(\sum_g \left[P_g^{\min} u_{g,t}^* + p_{g,t}^*\right] - f_g^*\right)$
  $\rho^{k+1} \leftarrow Newton(\rho_k, [g_i]_{i=0}^k)$
  if $L_k \geq Lowerbound$ **then**
    $Lowerbound = L_k$
  if $||\rho^k - \rho^{k+1}||_2 \leq \epsilon$ **then**
    break

Algorithm 2 $Newton(\rho, [g_i]_{i=0}^k)$

Let $\rho^{\max}$ and $\rho^{\min}$ be the the initial box for ρ
Let $a = [-g_{0:k}^T; I; -I]$ and $b = [\langle -g_i, \rho_i\rangle \ i = 0..k; \rho^{\max}\mathbf{1}; -\rho^{\min}\mathbf{1}]$
$x_0 \leftarrow SMP(a, b, \rho_k, g_k)$
while $||x_{k+1} - x_k||_2 \geq \epsilon$ **do**
  $F' \leftarrow \sum_i (b_i - \langle a_i, x\rangle)^{-1} a_i$; $F'' \leftarrow \sum_i (b_i - \langle a_i, x\rangle)^{-2} a_i a_i^T$
  $x_{k+1} \leftarrow x_k - [F'']^{-1} F'$

Theorem 8. *Let $\hat{L}(\rho, k)$ be the model function such as defined at* (3.8)*. Then*

$$\hat{L}(\rho, k) \geq L(\rho) \qquad \forall k$$

In order words, our piecewise linear function $L(\rho)$ is upper-approximated at each iterate by a model function $\hat{L}(\rho, k)$. At iteration 0, this is a single hyperplane. Then as the iterate k is growing, we are specifying our model function $\hat{L}(\rho, k)$, making it closer and closer to our target function $L(\rho)$ where it is interesting to, i.e. in the neighbourhood of the optimum.

This model function define the *master* program :

Master program

$$\begin{aligned} \max_{\rho\in Q,\theta} \quad & \theta \\ s.t. \quad & \theta \leq \langle a_i, \rho\rangle + b_i \quad \forall i = 0..k \end{aligned} \tag{3.9}$$

where $a_i = g_i$ is the "cut coefficient" and $b_i = L(\rho_i) - \langle g_i, \rho_i\rangle$ is the "cut constant".

Let's notice that solving such model function maximization program is straightforward as it is a classic linear optimization which can be solved using *simplex* or *interior-point* algorithms.

It remains to define how the iterate sequence $\{\rho_k\}_{k=0}^{\infty}$ should be built. The more intuitive way to build such sequence is choosing

$$\rho_{k+1} = \arg\max_{\rho} \hat{L}(\rho, k). \tag{3.10}$$

i.e. the solution of the master program (3.9). This defines the classical Kelley's cutting plane method.

A substantial advantage of schemes based on model function is that it explicitly provides an upper bound as well as a lower bound at each iterate. Indeed, in the context of the maximization problem (3.3), a lower bound at iterate k is defined as $LB_k = \min_{i=0..k} L(\rho_i)$. And from theorem 8, an upper bound at iteration k is

$$UB_k = \max_{\rho} \hat{L}(\rho, k) \qquad \forall k \tag{3.11}$$

Theorem 9. *The upper bounds sequence $\{UB_i\}_{i=0}^{k}$ is decreasing.*

Proof. This results is straightforward from the definition of the model function which leads to $\hat{L}(\rho, k+1) \leq \hat{L}(\rho, k) \quad \forall k.$ □

A graphical representation of Kelley's scheme applied on example 6 is provided at figure 3.4e.

Example 7. *Let's illustrate the concept of "model" function composed by supporting hyperplanes. The data used are those of example 2 in chapter 1, with a demand of 65 [MW]. Figure 3.5 illustrates the process.*

The initial price is 7. For such price, the oracle provides us a supgradient for building cut 1. Maximizing this single hyperplane function make the optimum price to jump at the extreme point of our boxed domain, in this case 40. Evaluating the supgradient at 40 provides cut 2. The optimum solution at this stage is the intersection of cut 1 and 2, around 17. Again, a supgradient is provided and cut 3 is built. The new optimum point is located around 24 at the intersection of cut 2 and 3. A supgradient is provided and cut 4 is built. The new optimum point is located at 22.5 at the intersection of cut 3 and 4. Evaluating the supgradient at this point gives the same cut as 3, so we conclude that 22.5 is indeed the convex hull price.

Let's go back to chapter 1. At example 3, a graphical analysis of the same study case sustained that 22.5 was the convex hull price for a 65 [MW] demand. It was theoretically established that solving (3.3) *is equivalent to finding the convex hull price. This example combined with example 3 illustrates such theoretical fact on a small study case.*

As illustrated in the last example, Kelley's algorithm in our case is finite. This is due to theorem 6 : as each iterate adds a new hyperplane and as the number of hyperplanes supporting the function is finite, the algorithm is finite. But despite its simplicity and good behaviour in low dimension, this scheme tends to be unstable as it makes big moves from iterate to iterate which leads to a poor convergence behaviour in higher dimensions.

3.3.3 Level cutting plane method

This method is a *regularization* of the previous scheme to tackle its unstable behaviour. Indeed, the Kelley's cutting plane method makes "big moves" at each iterates (see figure 3.4e). This is due to the unstable nature of piecewise linear functions : adding a single supporting hyperplane of the

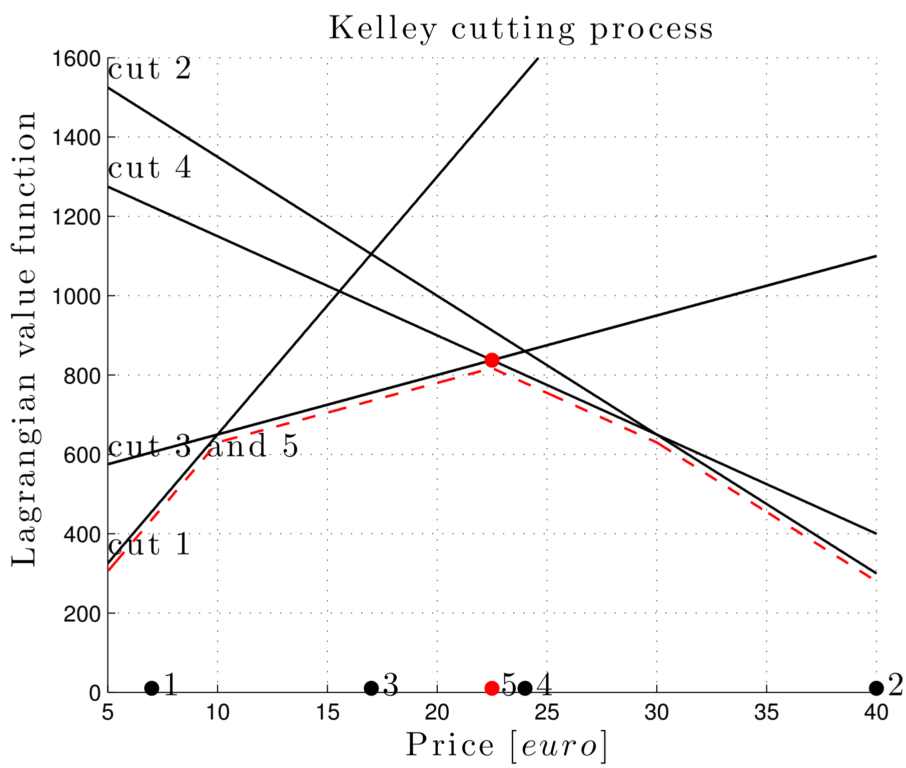


Figure 3.5: Kelley cutting plane method applied to a single period case, using the data of table 1.2 in chapter 1 with a load of 65 [MW]. The optimum found by the method is $\rho^* = 22.5$ [€/MWh]. The model function of the Lagrangian function at the end of the process is given in red.

function can move the optimum far from the previous point. In low dimension (one to five prices), it is not a concern. But increasing the dimension makes this method practically inefficient.

An improving idea, based on [8], would be to update the price more gently : instead of taking the optimum of the model function as the next iterate, let's chose ρ_{k+1} such that it is "better" than the current value regarding the model function $\hat{L}(\rho_{k+1}, k)$ without being optimum at all costs. This is done using the information of the upper and lower bound. More specifically, the new price ρ_{k+1} is chosen as the projection of ρ_k on the "level-set" $\hat{L}(\rho, k) \geq \alpha UB_k + (1-\alpha)LB_k$. This is solving

Price projection program

$$
\begin{aligned}
\min_{\rho \in Q} \quad & ||\rho - \rho_k||_2^2 \\
s.t. \quad & \langle a_i, \rho \rangle + b_i \geq \alpha UB_k + (1-\alpha)LB_k \quad \forall i = 0..k
\end{aligned}
\tag{3.12}
$$

where $a_i = g_i$ is the "cut coefficient" and $b_i = L(\rho_i) - \langle g_i, \rho_i \rangle$ is the "cut constant".

where $\alpha \in [0, 1]$ is a given parameter used to compute the average of UB and LB, i.e. the level set on which the price is projected. This is a quadratic program which can be solved very fast using `CPLEX`. To get rid of any doubt about such program, it is proven in [8] that a level set of a convex function is convex and a that a projection on a convex set exists and is unique. A graphical illustration of such projection process in 1-D is presented at figure 3.6.

Figure 3.4f shows the Level path of this method applied on example 6. As expected, the "big moves" encountered by Kelley's method are substantially reduced making the process more stable.

The Level algorithm is formally stated at algorithm 3.

Algorithm 3 *Level*$(\rho^0, \epsilon, iter^{\max}, \alpha)$

let $[g_i]_{i=0}^k$ be the supgradient at $[\rho^i]_{i=0}^k$; L_k be the Lagrangian function value at ρ^k; the lower bounds $LB \leftarrow -\infty$; the upper bounds $UB \leftarrow +\infty$; the cut coefficients $[a_i]_{i=0}^k$; the cut constants $[b_i]_{i=0}^k$

for $k \leftarrow 0, iter^{\max}$ **do**

  solve profit maximization (3.4) for each generator

  $g_k \leftarrow \left(D_t - \sum_{g \in G} \left[P_g^{\min} u_{g,t}^* + p_{g,t}^*\right]\right)$

  $L_k \leftarrow \sum_t \rho_t^k D_t - \sum_t \rho_t^k \left(\sum_g \left[P_g^{\min} u_{g,t}^* + p_{g,t}^*\right] - f_g^*\right)$

  $a_k \leftarrow g_k$; $b_k \leftarrow L(\rho_k) - \langle g_k, \rho_k \rangle$

  if $L_k \geq LB$ **then**

    $LB = L_k$

  solve the master (3.9)

  $UB \leftarrow \theta^*$

  solve the projection (3.12). Let ρ^* be the optimum.

  $\rho_{k+1} \leftarrow \rho^*$

  if $\frac{UB - LB}{|UB|} \leq \epsilon$ **then**

    break

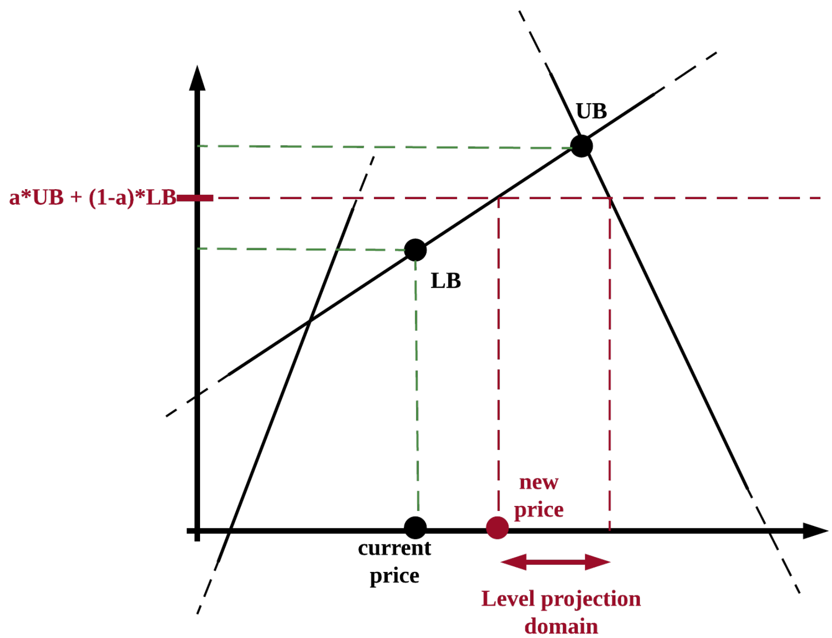


Figure 3.6: Level projection process. In this situation, some cuts have been built such as in Kelley method (e.g. figure 3.5). A lower bound LB is available as well as an upper bound UB. The "level projection domain" is defined as the price such that the model function (so the supporting hyperplanes) evaluated at these prices is greater or equal to $\alpha UB + (1-\alpha)LB$ (up the red dot horizontal line). The new price is the projection of the current price on this domain.

3.4 Stopping criterion

So far, not much has been said about the *stopping criterion*. At ACCPM algorithm 1, $||\rho^k - \rho^{k+1}||_2 \leq \epsilon$ has been used. However, in certain cases, this may lead to non-sense results. A theoretical stopping criterion would be $|L(\rho_k) - L^*| \leq \epsilon$ or $||\rho^k - \rho^*||_2 \leq \epsilon$. But as these quantities are not known *a priori*, it is practically useless.

As we are dealing with a maximization program, it is straightforward that a lower bound is simply the lowest evaluated value of the Lagrangian function. Defining an upper bound is less trivial. However, it has been stated that using model function approximation provides a convenient way of finding an upper bound, such as defined at (3.11). Such upper bound is useful as it can also help to define a stopping criterion. Indeed, one can define a more wise stopping criterion as the *relative gap* between the upper and lower bounds :

$$\frac{UB - LB}{|UB|} \leq \epsilon \tag{3.13}$$

As in the end, the target quantity is the price, it would be interesting to know what is the link between this stopping criterion and the price. Considering the expression of the Lagrangian function (3.3) :

$$\frac{UB - LB}{|UB|} \sim \frac{\langle \rho_1, D\rangle - \langle \rho_2, D\rangle}{|\langle \rho_1, D\rangle|} \sim \frac{||\rho_1 - \rho_2||}{||\rho_1||}. \tag{3.14}$$

Thus this stopping criterion is consistent with the accuracy we want to obtain on the price.

Finally, let's notice that such upper bound and stopping criterion can be incorporated in the ACCPM algorithm without damaging its computational performances as solving a small linear program is cheap.

3.5 Comparaison and Results

This section has four goals. (1) To calibrate the stopping criterion parameter ϵ. (2) To calibrate the α parameter of the Level method. (3) To test the computational and convergence behaviour of the schemes presented before, focussing on Level, Kelley, ACCPM and subgradient methods. (4) To analyse the robustness of the solution.

Let's keep in mind that our practical target is to solve the convex hull price on a 72 hours frame. As the profit maximization problems can be solved separately, the algorithms have been implemented using multiprocessing in Python.

Example 8. *For the following tests, a single node study has been built. The program solved is model* (2.2)*. There are 62 generators composed by nuclear, gas, biomass and oil power plants which consists of the Belgian power facilities. The data has been partially provided by the CREG while the missing ones are based on the study [14].*

The demand is computed as the real Belgian demand encountered some days of year 2014 lowered by the wind, solar and hydro production forecasts (available on ELIA website).

This is the study case for most of presented tests of this chapter unless explicitly indicated differently (e.g. some tests made on a 5 or 20 generators case).

3.5.1 Calibration of the ϵ-tolerance

Let's consider the stopping criterion as defined at (3.13). The major concern here is the choice of ϵ. As we are dealing with real problems with real data, the accuracy of the result is of course a sensitive question but it should not be pushed out of the limits of a reasonable expectation of such theoretical model on predicting the real market price[4]. In other words, it would be none-sense to compute the result with a precision that completely overpasses the inherent precision of the data or the expected "noise" in the system endured by any real applications.

Figure 3.7 shows the results of the Level method and the ACCPM method. The "real" CHP (the black curve) was obtained by the Level method, pushing ϵ to a very low value. A 1% relative gap seems too weak as the differences are still dramatics. A 0.1% relative gap performed better but as it could depend from case to case, a relative gap of 0.05% looks safer. This is the value used in the rest of the tests.

Let's notice that the Lagrangian function seems to be quite flat, which results in possibly many "ϵ-optimum convex hull prices".

ϵ selected choice

$$\epsilon = 0.0005 \text{ (i.e. } 0.05\%) \tag{3.15}$$

3.5.2 Calibration of the α parameter

Regarding the Level algorithm, one question remains : to pick the right value α for (3.12). $\alpha = 1$ corresponds to the classic Kelley's method. On the other hand if $\alpha = 0$ then the price does not move. The challenge is to find an effective α. Figures 3.8 shows empirical results of the Level method on different cases in terms of the α parameter. Figure 3.8a highlights the fact that in very low dimension Kelley's method works well (the best α tends to 1). But as the dimension increases, it is more effective to use lower value of α. Figures 3.8b and 3.8c indicates that for 72 hours cases, the best α seems to be

Best α-parameter

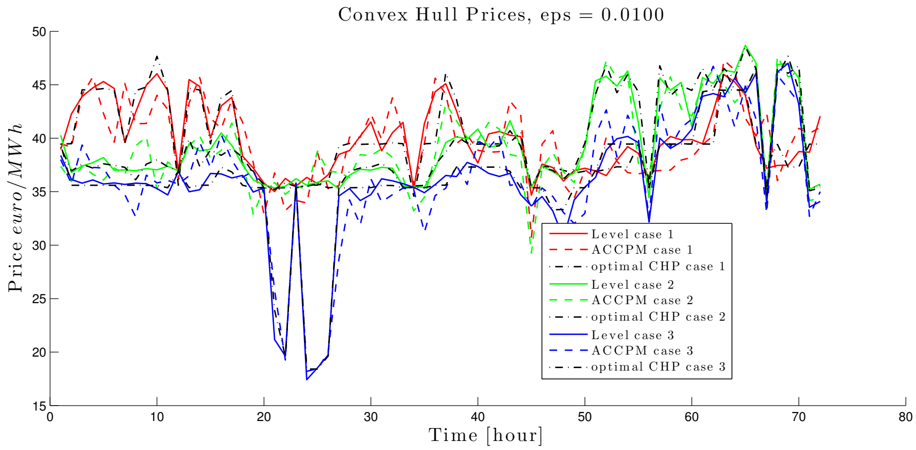
$$\alpha^{best} \approx 0.2. \tag{3.16}$$

In the rest of the material, unless it is specified differently, it is the value used.

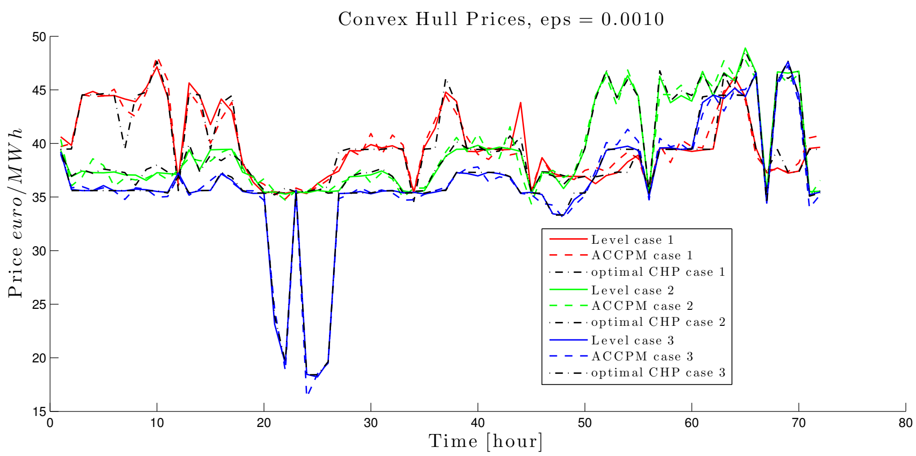
3.5.3 Convergence behaviour

First of all, let's compare a bit some previous methods on a single 72 time period example. Figure 3.9 compares the main algorithmic schemes presented previously. The Level method is definitely the most promising one. On the contrary, Kelley's method do not move from the initial gap. This is due to the fact that the method makes such great moves that it does not approach the optimum,

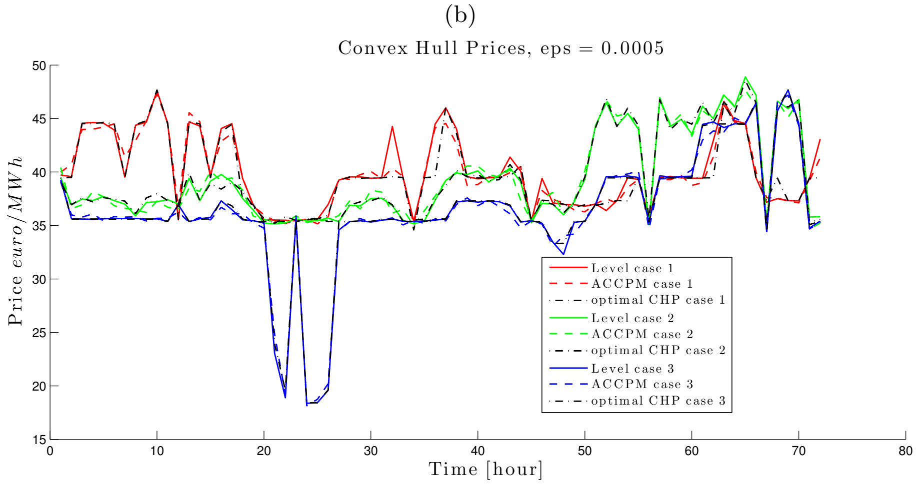
[4]Of course, this depends on the goal and the application. As here (in Europe) the objective is mainly to *predict* the price of the market with this *fundamental* model, we are more looking for an ϵ-convex hull price than the "exact" convex hull price. If the goal was to use this work as a model for *computing* the price of the market (e.g. in USA), then having a greater accuracy would sound reasonable.



(a)

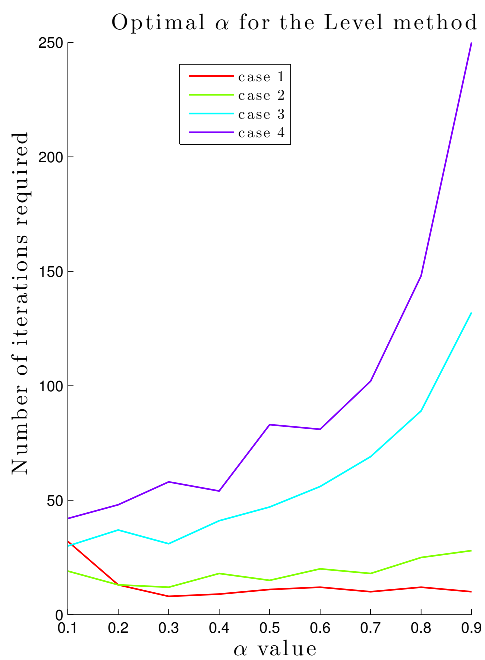


(b)

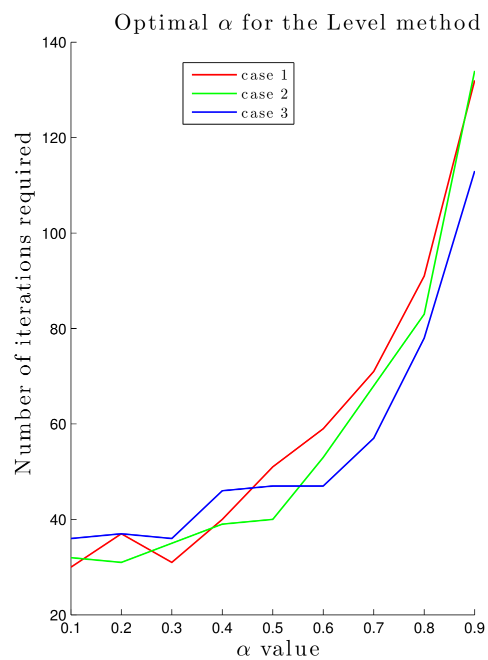


(c)

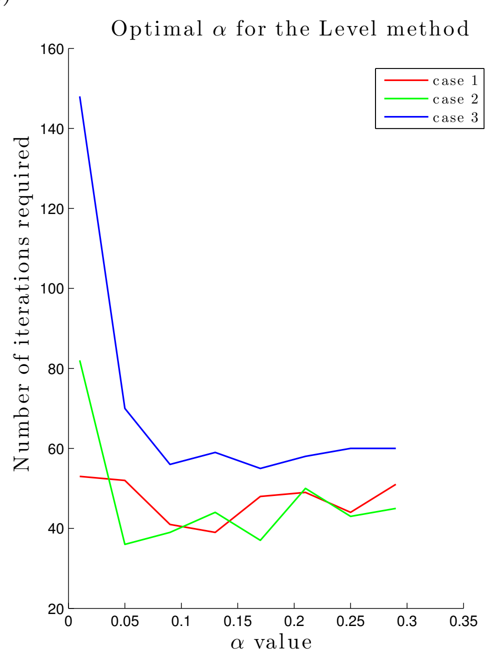
Figure 3.7: Convex hull price obtained by the Level method and the ACCPM method with the precision ϵ. The study case is as detailed at example 8 with 72 hours. The three cases correspond to three different shapes of demand. 3.7a corresponds to $\epsilon = 0.01$; 3.7b corresponds to $\epsilon = 0.001$; 3.7c corresponds to $\epsilon = 0.005$.



(a)



(b)



(c)

Figure 3.8: Computational effort of the Level method depending of the α parameter. 3.8a studies 4 cases corresponding to 4 different dimensions of the problem, respectively 2, 5, 24 and 72 hours. 3.8b and 3.8c study three 72 hours cases with a different shape of demand curve.

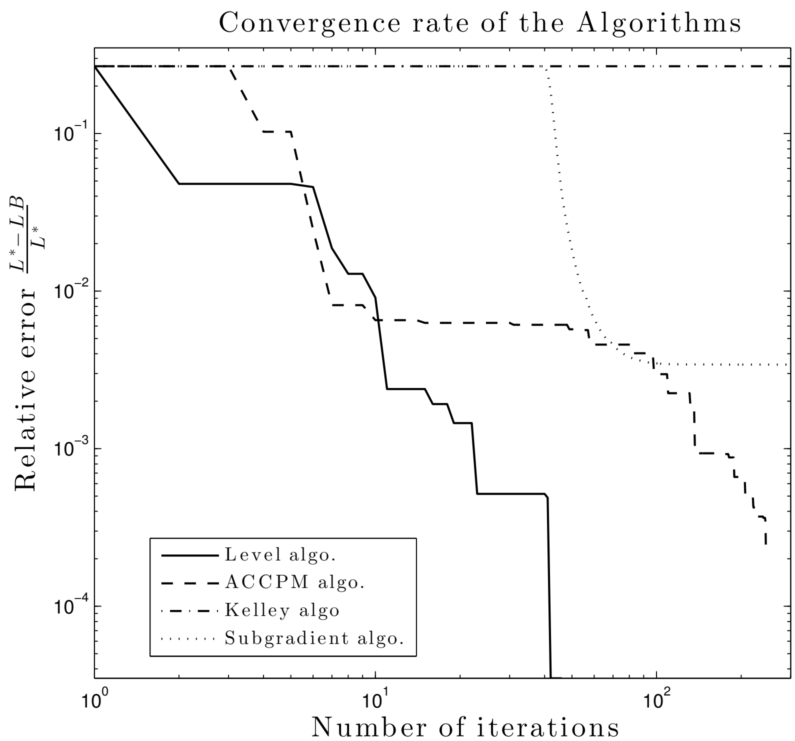


Figure 3.9: Relative error ($\frac{L(\rho^*)-LB}{L(\rho^*)}$) of the Level, Kelley, ACCPM and subgradient algorithms. The test case is the one of example 8 with 72 hours. Pay attention to the y-axis : as no UB is computed in the subgradient method, a first run was made to obtain the convex hull price ρ^* with the corresponding value function L^* and it is this value which is used, i.e. $\frac{L^*-LB}{|L^*|}$.

therefore, the lower bound is not improved at all (the upper bound is but is not used to plot this graph).

The CPU time is not the most accurate measure of convergence behaviour as it depends on the computer devices. However, in order to give a picture of how long does such simulation take, table 3.1 provides the CPU times on a 72 hours example.

In order to highlight the trends, three sets of tests have been built :

- The first one makes the number of hours to vary (the dimension of the problem). This is illustrated at figure 3.10. It is pretty clear that the number of iterates increases with the dimension (this is expected, considering the theoretical rates of convergence of the ACCPM, Kelley and Level method, see [8]). For the Kelley's algorithm, it is dramatic. Indeed, in low dimension (only 2 or 5...) it works well. But in higher dimension (24, 48 or 72) it completely collapses. ACCPM works well even in higher dimension. But the Level method clearly surpasses ACCPM.

- Then a second test makes the number of generators to vary and measure the impact of such variation on the number of iterations. This is illustrated at figure 3.11. It is pretty clear that

	Level method	ACCPM
Iterations	37	99
CPU[s]	163	343

Table 3.1: The results have been performed on a 72 hours case, with 62 generators such as described at example 8 and for which the convergence rate has been displayed at figure 3.9. The algorithms ran on a Intel Core i5 processor using python multiprocessing on 4 cores.

this does not impact at all the number of iterations required to converge (but it impacts the CPU time as the time to obtain the supgradient increases). Level method is definitely more effective than ACCPM method.

- In order to be sure that such promising behaviour of the Level method is not just coincidence due to the specific example used, a last test has been built which makes the shape of the demand to vary. This is illustrated at figure 3.12. Clearly, the number of iterates required to converge is similar from one case to another and the Level method clearly performs better than ACCPM.

3.5.4 Robustness : volatility of the iterate sequence

As the pricing of electricity is a sensitive question which requires constant resolution, a major concern is the robustness of the algorithmic scheme. The tests presented here analyse the volatility of the prices iterates sequence. In order words the variance of the iterates sequence $\{\rho_i\}_{i=0}^k$ generated by the algorithm. Figure 3.13 provides box plots (one box for each hourly price $\rho_t^k,\ t \in T$, as this is a 24 hours example there are 24 boxes) of the price sequence obtained by the Level, Kelley and ACCPM algorithm.

As a reminder, the box plot is composed as follow. The box is divided in three parts: the lower part is the first quartile (the lower black line), the center part is the second and third quartile (the blue box), the upper part is the fourth quartile (the upper black line) while the red crosses are the "outliers".

Clearly as much as the box is big and the number of outliers is large, as much the iterates oscillate. As sustained by the theory, the iterates provided by Kelley's algorithm are quite unstable and oscillate a lot. ACCPM does a greater job but is still outperformed by the Level method. This means that, after a very few iterates, the ρ_k are already pretty closed to the CHP ρ^*. Or saying it differently, stopping the algorithm before the end of the process would not provide a non-sense price as it would be the case with Kelley.

3.5.5 Conclusion of the tests

The tests clearly demonstrate the very promising behaviour of the **Level algorithm**. Level algorithm is the most efficient on each and every examples. Furthermore, it does not seem to explode as the dimension increases. Finally it is clearly the most robust method. **It is the algorithmic scheme retained for the upcoming tests in chapter 4 and seems to be the most promising one for real industrial applications.**

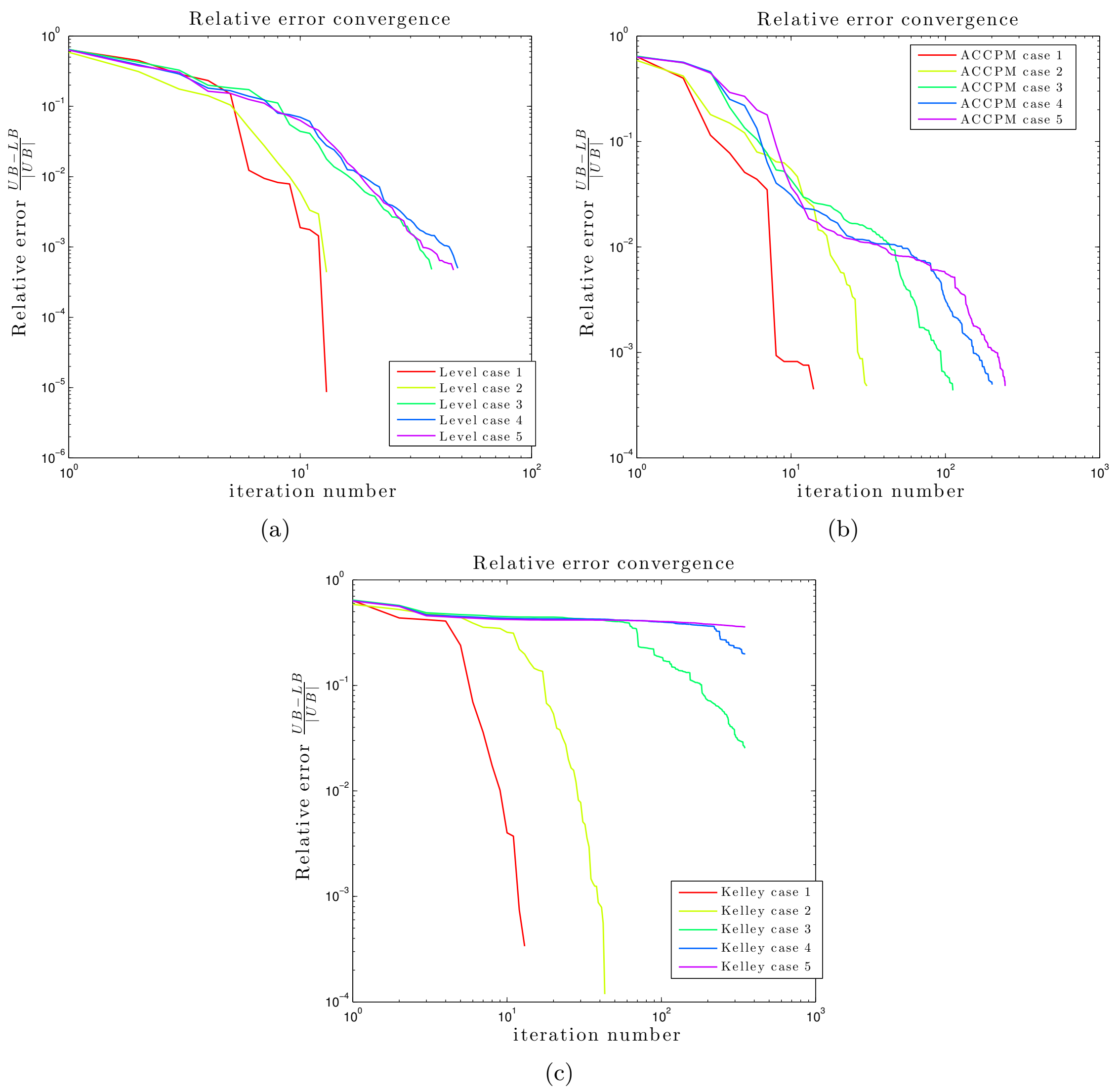
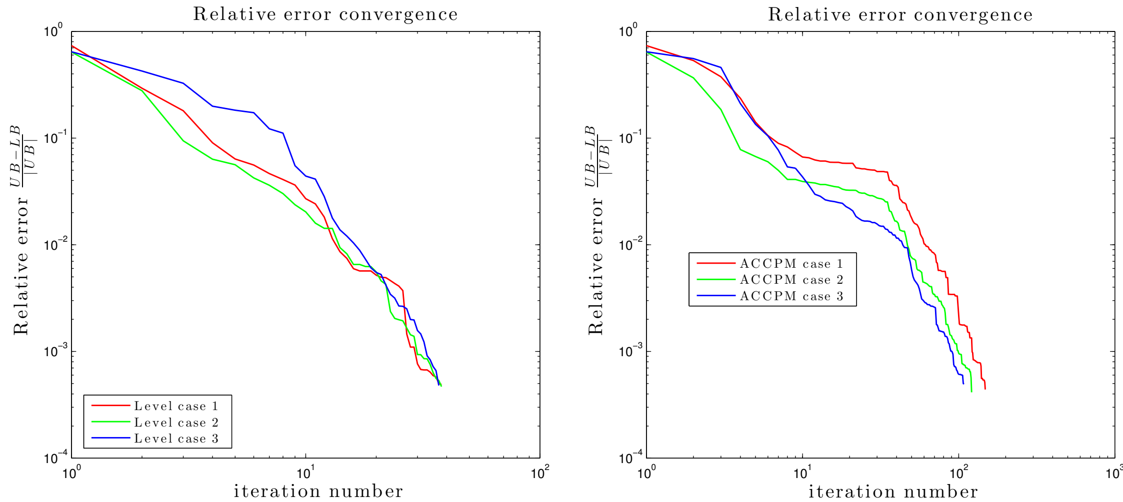
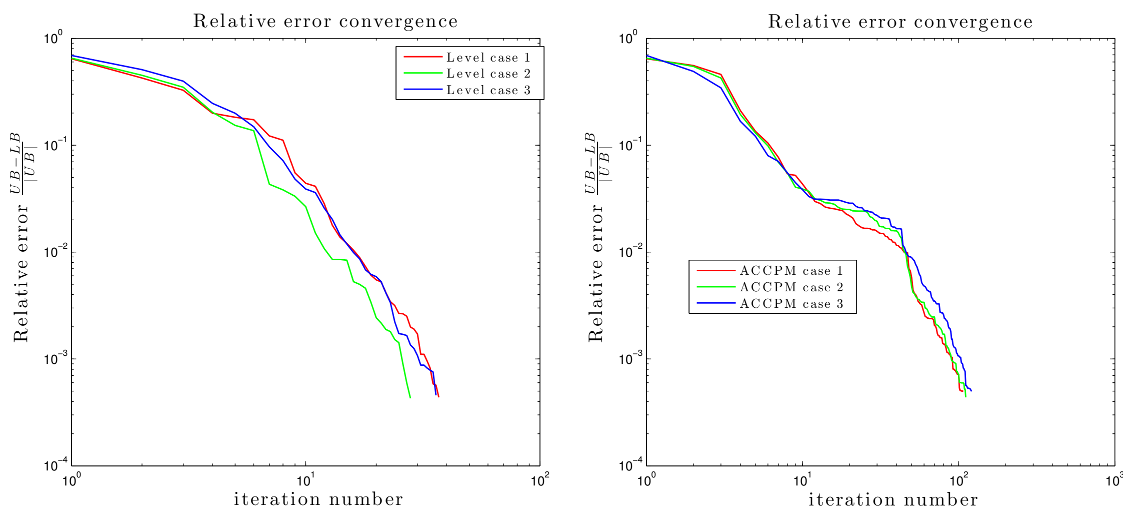


Figure 3.10: Convergence depending on the number of hours. Relative error of the Level, ACCPM and Kelley's algorithm in terms of the iterates. These are 62 generators study cases of example 8. Cases 1 to 5 correspond respectively to a 2, 5, 24, 48 and 72 hours case.



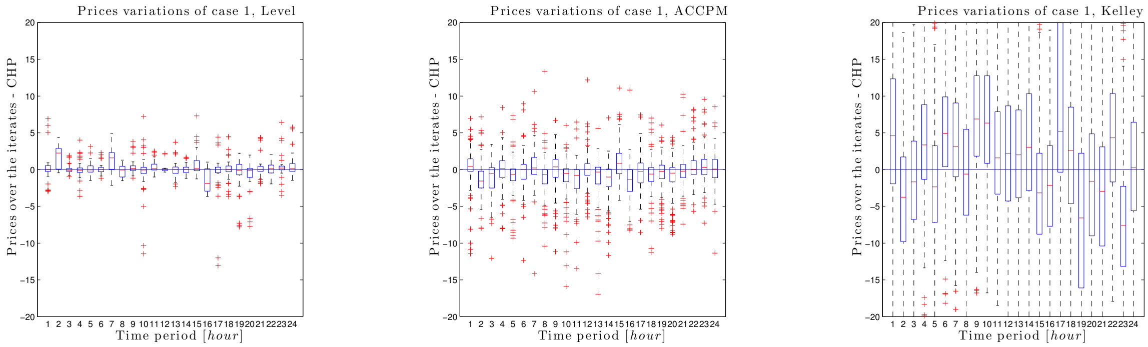
(a) (b)

Figure 3.11: Convergence depending on the size of the generator set. Relative error of the Level and ACCPM algorithm in terms of the iterates. These are 24 time periods study cases. Case 1 corresponds to a 5 generators case (adapted from example 8); case 2 corresponds to a 10 generators case (adapted from example 8) and case 3 to 62 generators case (example 8).

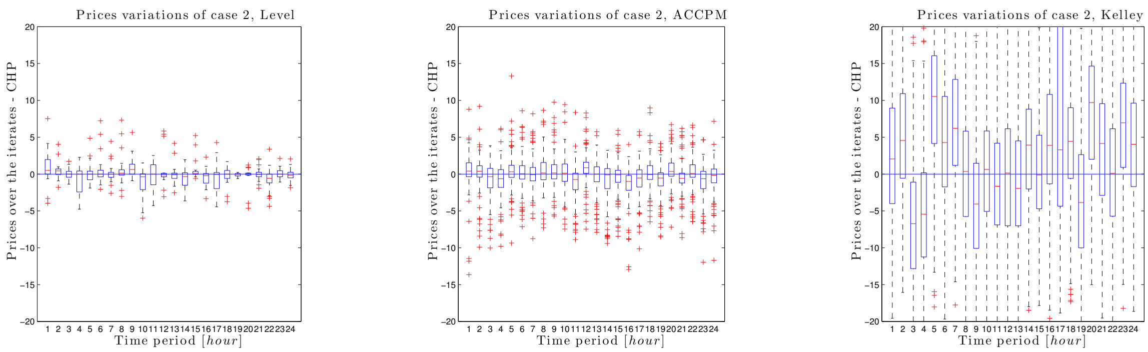


(a) (b)

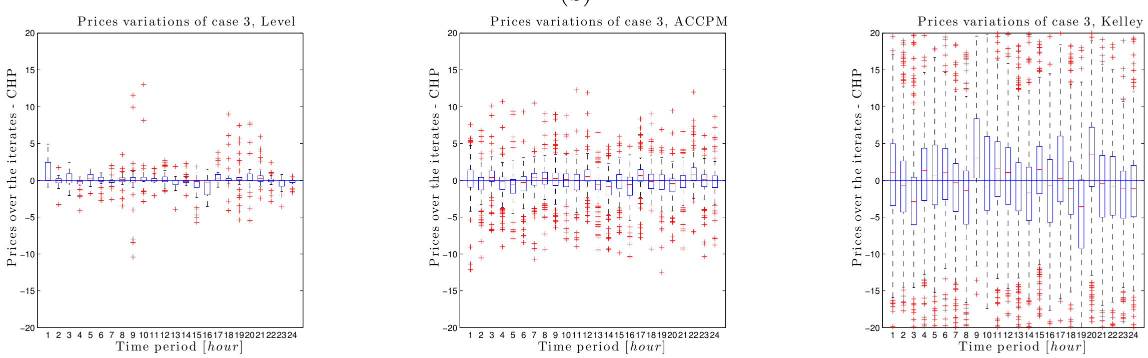
Figure 3.12: Convergence depending on the shape of the demand. Relative error of the Level and ACCPM algorithm in terms of the iterates. These are 62 generators study cases of example 8 on 72 hours (as it is our practical target). Cases 1 to 3 corresponds to three different shapes of demand encountered in Belgium during the year 2014.



(a)



(b)



(c)

Figure 3.13: Illustration of the volatility of the iterate sequence obtained (from left to right) by the Level, the ACCPM and Kelley's algorithm. The tests have been made on a 62 generators with 24 hours.

3.6 Convex parts and dualization

Let's complexity the model (3.3). Let's suppose that some generators are convex (no binary on/off decisions, no start-up costs, no no-load costs...), which often arises in practical cases. Then the model could be written as :

$$\max_{\rho} \left\{ \sum_t \rho_t D_t - \sum_{g \in G^{conv}} \max_{\widehat{p} \in X_g^{conv}} \left\{ \sum_t \rho_t \widehat{p}_{g,t} - f_g \right\} - \sum_{g \in G^{bin}} \max_{\substack{u,v,w,p,\\ \delta \in X_g}} \left\{ \sum_t \rho_t \left[P_g^{\min} u_{g,t} + p_{g,t} \right] - f_g \right\} \right\} \tag{3.17}$$

Where the convex profit maximization program of the convex generators is

$$\begin{aligned} \max_{\widehat{p}} \quad & \sum_g \sum_t \rho_t \widehat{p}_{g,t} - C_g^P \widehat{p}_{g,t} & \\ s.t. \quad (\nu_{g,t}) \quad & \widehat{p}_{g,t} \le P_g^{\max} & \forall g,t \\ (\mu_{g,t}) \quad & \widehat{p}_{g,t} \ge P_g^{\min} & \forall g,t \end{aligned} \tag{3.18}$$

A way of dealing with these parts would be to treat them just as the non-convex parts, i.e. to solve them given a price ρ_t and generate supgradient... as explained in the beginning of this chapter. However, as these parts are convex, there might be a more wise way to deal with it. As the problem is convex, the dual of (3.18) can be stated as :

$$\begin{aligned} \min_{\nu,\mu} \quad & \sum_g \sum_t \nu_{g,t} P_g^{\max} - \mu_{g,t} P_g^{\min} & \\ s.t. \quad & \rho_t - C_g^P + \mu_{g,t} - \nu_{g,t} = 0 & \forall g,t \\ & \nu_{g,t}, \mu_{g,t} \ge 0 & \forall g,t \end{aligned} \tag{3.19}$$

Considering the fact that $-\min_x f(x,y) = \max_x -f(x,y)$ and that $\max_y \max_x f(x,y) = \max_{x,y} f(x,y)$, the dual (3.19) of the convex generators can be incorporated in (3.17) as :

$$\begin{aligned} \max_{\rho,\mu,\nu} \quad & \sum_t \rho_t D_t - \sum_{g \in G^{conv}} \sum_t \left(\nu_{g,t} P_g^{\max} - \mu_{g,t} P_g^{\min} \right) - \sum_{g \in G^{bin}} \max_{\substack{u,v,w,p,\\ \delta \in X_g}} \left\{ \sum_t \rho_t \left[P_g^{\min} u_{g,t} + p_{g,t} \right] - f_g \right\} & \\ s.t. \quad & \rho_t - C_g^P + \mu_{g,t} - \nu_{g,t} = 0 & \forall g,t \\ & \nu_{g,t}, \mu_{g,t} \ge 0 & \forall g,t \end{aligned} \tag{3.20}$$

Meaning that the duals variables are now explicitly variables of the "master program".

If any doubt subsists about how the previous algorithmic schemes could be used to tackle such problem, let's write the master program (3.9) adapted with (3.20) :

Adapted Master program

$$\begin{aligned}
\max_{\substack{\rho\in Q,\theta\\ \nu,\mu}} \quad & \sum_t \rho_t D_t - \sum_{g\in G^{conv}} \sum_t \left(\nu_{g,t} P_g^{\max} - \mu_{g,t} P_g^{\min}\right) - \theta \\
s.t. \quad & \theta \geq \langle a_i, \rho\rangle + b_i && \forall i = 0..k \\
& \rho_t - C_g^P + \mu_{g,t} - \nu_{g,t} = 0 && \forall g,t \\
& \nu_{g,t}, \mu_{g,t} \geq 0 && \forall g,t
\end{aligned} \tag{3.21}$$

where $a_i = \sum_{g\in G^{bin}} P_g^{\min} u_{g,t} + p_{g,t}$ is the "cut coefficient" and $b_i = -\sum_{g\in G^{bin}} f_g$ is the "cut constant".

Where the "cuts" are only made on the non-convex generators profit maximization function of value θ. The "projection program" (3.12) of the Level method, addapted with (3.20) is :

Adapted projection program

$$\begin{aligned}
\min_{\substack{\rho\in Q,\nu\\ \mu,\theta}} \quad & ||\rho-\rho_k||_2^2 + ||\mu-\mu_k||_2^2 + ||\nu-\nu_k||_2^2 \\
s.t. \quad & \sum_t \rho_t D_t - \sum_{g\in G^{conv}} \sum_t \nu_{g,t} P_g^{\max} - \mu_{g,t} P_g^{\min} - \theta \geq \alpha UB_k + (1-\alpha) LB_k && \forall i = 0..k \\
& \theta \geq \langle a_i, \rho\rangle + b_i && \forall i = 0..k \\
& \rho_t - C_g^P + \mu_{g,t} - \nu_{g,t} = 0 && \forall g,t \\
& \nu_{g,t}, \mu_{g,t} \geq 0 && \forall g,t
\end{aligned} \tag{3.22}$$

where $a_i = \sum_{g\in G^{bin}} P_g^{\min} u_{g,t} + p_{g,t}$ is the "cut coefficient" and $b_i = -\sum_{g\in G^{bin}} f_g$ is the "cut constant".

A question remains, how can the lower and upper bounds be computed? This question is not as obvious as previously, as other master variables and constraints have been added.

Given a testing price ρ one can solve the profit maximization to obtain a cut. However, It is not possible to conclude that it is a lower bound of the Lagrangian function. Indeed, the other components of the objective function depending on the "reactions" of the other master variables (ν and μ at (3.21)) are not known. In order to compute these values, it is necessary to solve the master program 3.21 but with the price fixed to ρ. So there is one more step in the algorithm (this is not damaging as the master is a linear program). The procedure is described mathematically at algorithm 4 and is illustrated schematically at figure 3.14.

Remark This part was focussed on processing the "convex generators" but the same reasoning could be applied on all the convex parts of the problem, e.g. the network constraints (PTDF), the pump-storage plant... This is detailed in appendix A, where the full model is presented.

Algorithm 4 $Level2(\rho^0, \epsilon, iter^{\max}, \alpha)$

the lower bounds $LB \leftarrow -\infty$; the upper bounds $UB \leftarrow +\infty$; the cut coefficients $[a_i]_{i=0}^k$; the cut constants $[b_i]_{i=0}^k$

for $k \leftarrow 0, iter^{\max}$ **do**

  solve profit maximization (3.4) for each generator. Let π^* be the total profit.

  $a_k \leftarrow \sum_{g \in G^{bin}} P_g^{\min} u_{g,t} + p_{g,t}$; $b_k \leftarrow -\sum_{g \in G^{bin}} f_g$

  solve the master (3.21), the price being fixed to ρ_k. Let L^* be the optimum objective.

  $L_k \leftarrow L^* + \theta^* - \pi^*$

  if $L_k \geq LB$ **then**

    $LB = L_k$

  solve the master (3.21). Let L^* be the optimum objective.

  $UB \leftarrow L^*$

  solve the projection (3.22). Let ρ^* be the optimum.

  $\rho_{k+1} \leftarrow \rho^*$

  if $\frac{UB-LB}{|UB|} \leq \epsilon$ **then**

    break

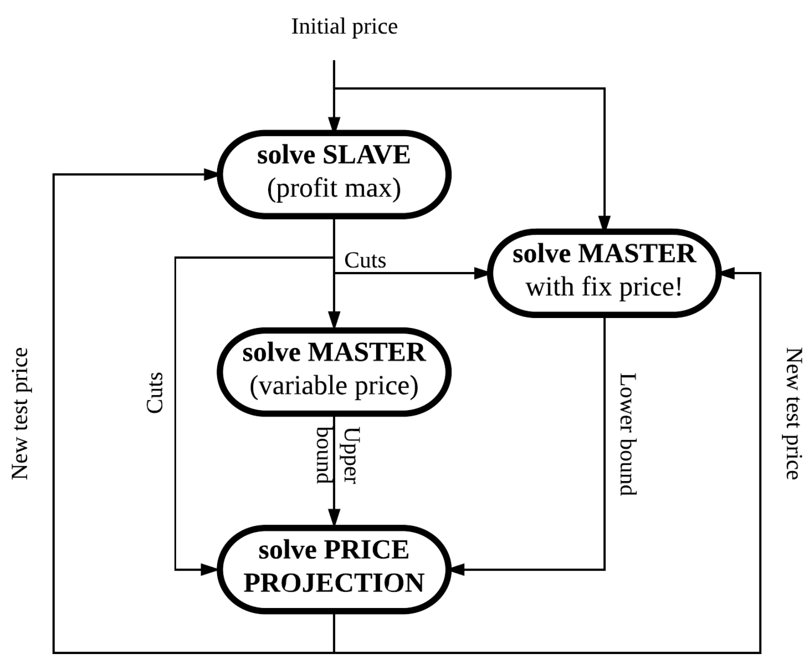


Figure 3.14

Chapter 4

The convex hull price for the European market

Chapter 1 brings a consistent economical model for pricing electricity in unit commitment. Chapter 2 provides an efficient and accurate way to model the profit maximization of non-convex plants. Chapter 3 comes up with an efficient algorithm for finding the convex hull price. Now we have everything we need to test it on a real study case, this is the subject of the present chapter.

4.1 The European market architecture

As suggested in chapter 1, the electricity market is complex. The objective of this section is certainly not to address its complexity in detail but to give a picture of the overall process as this is a basic knowledge useful in order to understand the study case presented in this chapter.

The actors There are basically three levels in order to bring electricity to the private consumers :

- the production, i.e. the nuclear, oil, gas... power plants;
- the transmission, i.e. the high-voltage lines used to transport electricity between nodes in the power system;
- the distribution, i.e. the smaller lines providing power everywhere

The price at which the private consumers buy electricity from the distributor is generally fixed at the year and is none of our interest here. We are interested in the "whole-sale" electricity market which consists of system operators which buy electricity from the producers in order to meet the demand.

Stages It has already been highlighted that the electricity market is highly volatile. This is partially due to a demands evolving constantly. As the demand changes, the production has to be adjusted. But what should be a wise time step? On the one hand, some units would need to start-up/down which is required to be known in advance. On the other hand, the time step should be small enough to capture the demand variations. Therefore, the market implies several stages. Let's simplify a bit and denote the "main" stages as :

- day-ahead market, which is run a day in advance based on the forecast of the demand. This is required as some units need informations in order to decide whether to start-up or shut-down. The typical time step is one hour.
- intra-day market, which is run during the day and is required in order to adjust what the day-ahead market did based on the demand forecast with the effective encountered demand. The typical time step is fifteen minutes.

This work is fully focussed on day-ahead hourly market.

Network The network is modelled as nodes and lines. A natural approach would consist of including the Kirchhoff's laws in order to model the physical constraints inherent to any electrical system. As it would be convenient to deal with linear optimization, a linear version of these laws, called direct current power flow equations are included. This leads to "nodal" pricing as each node could face a different price.

However, in the specific case of Europe, it is wished that a whole country, even if composed by several nodes, faces the same price. Therefore, nodes are aggregated in zones (countries) linked to each other by lines. As this is quite artificial and lines and zones do not have a straightforward physical meaning, the question of including Kirchhoff's laws remains. Without entering in any more details, it appears in practise that some lines are modelled using these laws (these are flow-based lines) while others only face limits in the energy they can transport (these are ATC lines).

Market architecture There are many possible ways to build a market. For the sake of simplicity, let's consider two possible designs :

- a pool : producers submit their technical constraints and the operator solve an optimization program such as (1.3) in order to commit units and dispatch power;
- an auction : producers submit bids of price-quantity and the operator buys them.

It is clear that the first one is more "centralized" whereas the second is "decentralized". Pricing model such as convex hull pricing could be used directly by the operator in a pool to **compute** the price. In the case of an auction, this model can be used in order to **predict** the auction price.

In Europe, the market is organized as an auction.

EUPHEMIA EUPHEMIA is the algorithm used to clear the auction. As a pool suggests very straightforward way for the producers to bid (they just provide their technical features), auction makes the task more difficult as they need to transform these technical constraints into simple bids of price quantity. For a comprehensive explanation of EUPHEMIA, we refer the reader to the public description[1]. Briefly, EUPHEMIA establishes several bidding rules. The simplest bids are the **hourly orders**, which are basically stepwise or piecewise linear functions of the price depending on the power they are asked to produce, for a given hour. These bids can be partially accepted. Due to the technical characteristics of some plants which are impossible to guarantee with hourly orders (e.g. ramping, fixed costs...), other bidding rules are available such as the **block orders** which consist of a "profile" of production/price over several hours. These bids must be fully accepted or rejected. A plant can submit several block orders and specify **linked blocks** (which links different

[1] "EUPHEMIA Public Description, PCR Market Coupling Algorithm" available on the web.

block orders to each other) and **exclusive blocks**. A last type of bidding rule are the **complex orders**.

Unlike the pool, such auction process does not correspond to a "fundamental" pricing model. The aim of the simulations presented in the next sections is to see how does a fundamental model such as convex hull pricing fit the price that comes out of the auction.

4.2 A model for Europe

4.2.1 Mathematical model

Let's detail the features of the complete model that was implemented. The mathematical formulations are fully provided at appendix A.

- **the network**
  - 17 countries are considered, i.e. at, be, ch, cz, de, dke, dkw, es, fr, hu, itnorth, nl, no, se4, si, uk and pl.
  - 6 of them (Austria, Belgium, Switzerland, Germany, France and Netherlands) are modelled using real plants informations installed in those countries while the others are modelled as a single generator enduring a variable revenue which is the real market price observed in the corresponding country.
  - Among these six countries, as some data were not available, a part of the produced energy is considered as exo-production (parameter instead of a variable). This is especially true for Austria and Switzerland. So in the end, Belgium, France, Germany and Netherlands are really the countries we are looking at and for which it is interesting to see the results, the others being there to avoid bad border effects. Belgium is certainly the most accurately modelled country.
  - The Central-West-European market (CWE, i.e. Belgium, France, Germany and Netherlands) is modelled using a flow-based approach as it is the case for real. All the other lines are ATC.
- **Generators**
  - There are about 400 generators.
  - CST, GT, CCGT and CGT are modelled as non-convex plants (involving binary on/off decision, start-up costs...) as it is those plants which are expected to provide the "flexibility" in the system, so for which these non-convex features are expected to be relevant. There are about 200 plants.
  - wind and solar power plants are modelled as exo-production, i.e. their production is supposed to be known based on forecast. There are about 25 plants.
  - the model includes one pump-storage facility, i.e. hydro power production involving pumping, realising water and storing water, the central of COO in Belgium.
  - all the other plants (nuclear, remaining hydro plants, oil...) are considered as convex pants (without start-up cost or other fixed costs). There are about 160 plants.
- **Generators features**

	1 core	3 cores
Average CPU time $[s]$	20	10

Table 4.1: Time for solving the profit maximization for all the non-convex generators of the complete model of section 4.2 on a 24 hours study case (about 200 plants).

  – for non-convex generators, the model as presented at chapter 2 is considered, with two differences : each plant has the possibility to use different fuels (this is modelled by adding one set of indices representing the fuels); the start-up cost does not varies in terms of the off time.
  – for convex generators, the multi-fuel case is also considered, but the only physical constraints are the generation capacity limits.

- **Cost structure** : in the previous chapters we used the notation C^P or C^{SU} to denote the production costs or start-up costs as it is more convenient. In practice, expressing the costs is a little more intricate. The efficiency of a plant can be defined as $\eta = \frac{ElectricEnergyOut}{ThermalEnergyIn}$. Then the heat rate is defined as $hr = \frac{1}{\eta}$. And the cost of producing power p with fuel f is $hr^{var} \times price^f$. In addition to that we also consider CO2 taxes which depend on the fuel type as well as other "variable" costs. The no-load and start-up costs are modelled similarly using "no-load heat rate" and "start-up extra fuel" parameters.

4.2.2 Implementation

The algorithm used to find the convex hull price is the Level method detailed at section 3.3.3, implemented in Python. As the profit maximization programs (program (3.4) which provides a subgradient for the algorithm) of each plant are independent, multi-processing computing has been used[2], and improves substantially the CPU time. Table 4.1 gives a picture of such trend. It should be noted that solving the profit maximization is the costliest part of each iterate, so multi-processing dramatically improve the total CPU time.

The models (master, price projection, profit maximization) are implemented in GAMS. As the algorithm requires to solve many times the same mathematical program but with very few changes in the parameters, "Gams Modifiers" have been used : the matrix of each program is generated at the beginning and then only the entries that have changed are modified in the matrix.

4.3 Results

4.3.1 Convex hull price for energy

In this section, a model **without** reserve is considered. The convex hull prices for the interested countries are illustrated at figures 4.1 and 4.2. The figures also include the "economic dispatch" (ED) price which is the price obtained by solving a classical economic dispatch considering all the plants as convex[3].

[2]It should be noted that multi-processing is not multi-threading. In fact multi-threading was not suitable for this application in Python. See Python documentation for more information.

[3]As a reminder, it has been highlighted in chapter 1 that a great property of the convex hull price is that when applied on a convex problem, it provides the same price than the marginal pricing.

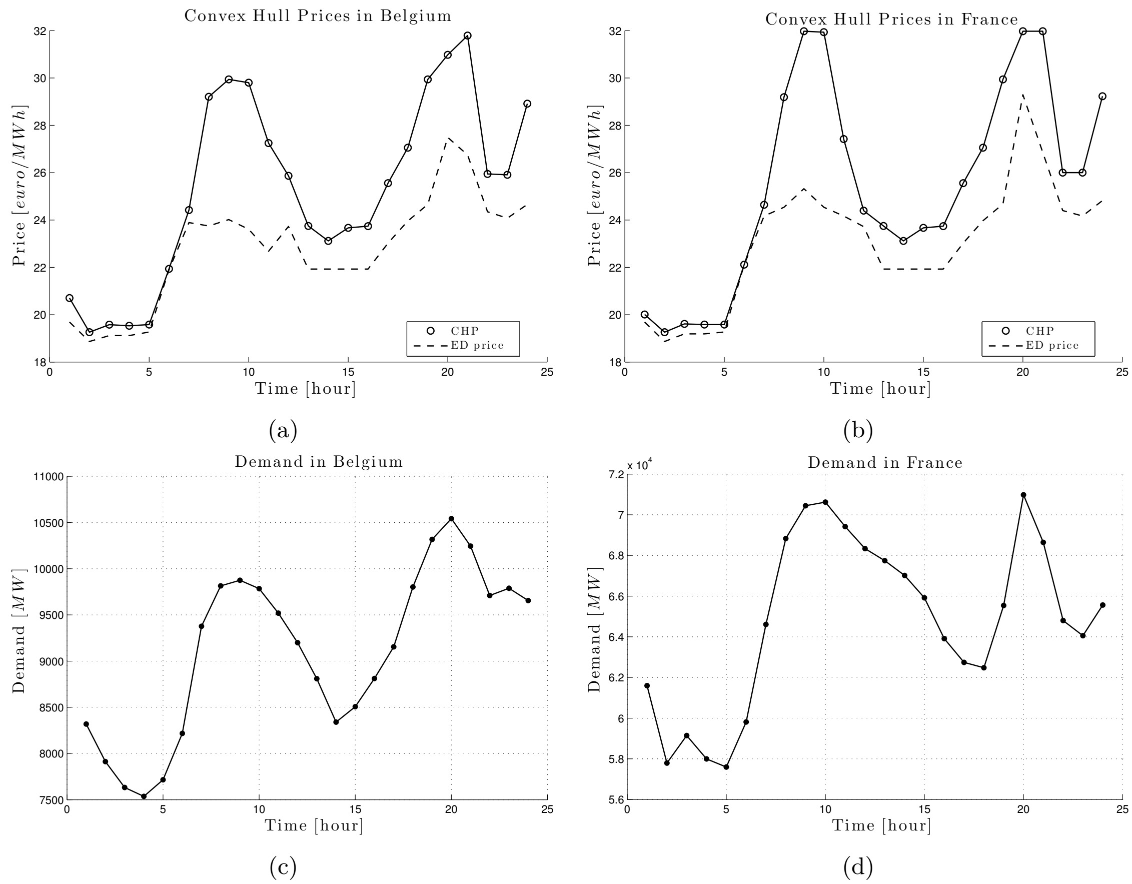


Figure 4.1: Convex hull price solving the model in Europe as described at section 4.2 with 24 hours. The left figure correspond to Belgium and the right figure to France.

Let's point out some observations :

- CHP and ED prices are quite similar during the night as at that time, the price is mainly driven by the convex components.
- During the day, CHP tends to shift the price a bit higher than what marginal pricing does.
- When a "peak" of demand occurs, CHP makes significant spikes of prices compared to ED prices. This is of course due to the start-up costs of non-convex units.

Figure 4.3 shows the production profile of four units (two ccgt, one cst and coo power plant) in reaction of the CHP in Belgium.

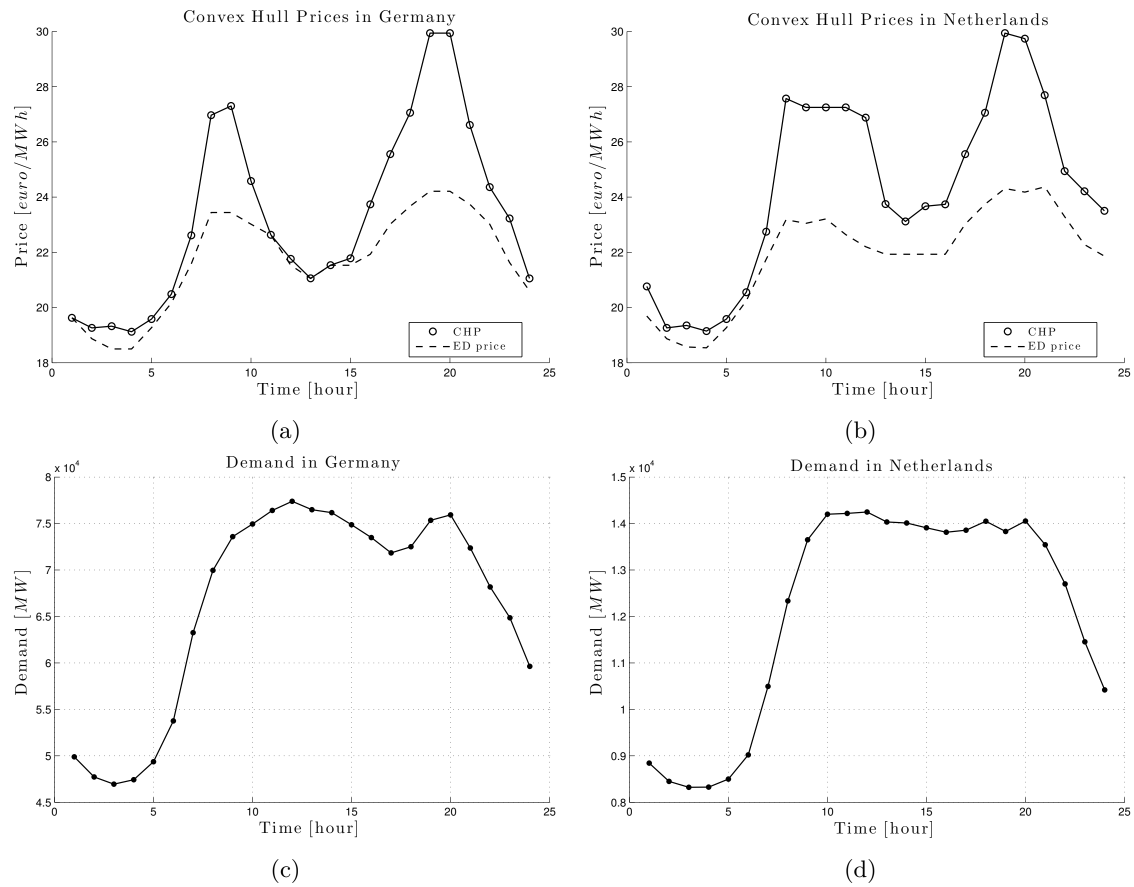


Figure 4.2: Convex hull price solving the model in Europe as described at section 4.2 with 24 hours. The left figure correspond to Germany and the right figure to Netherlands.

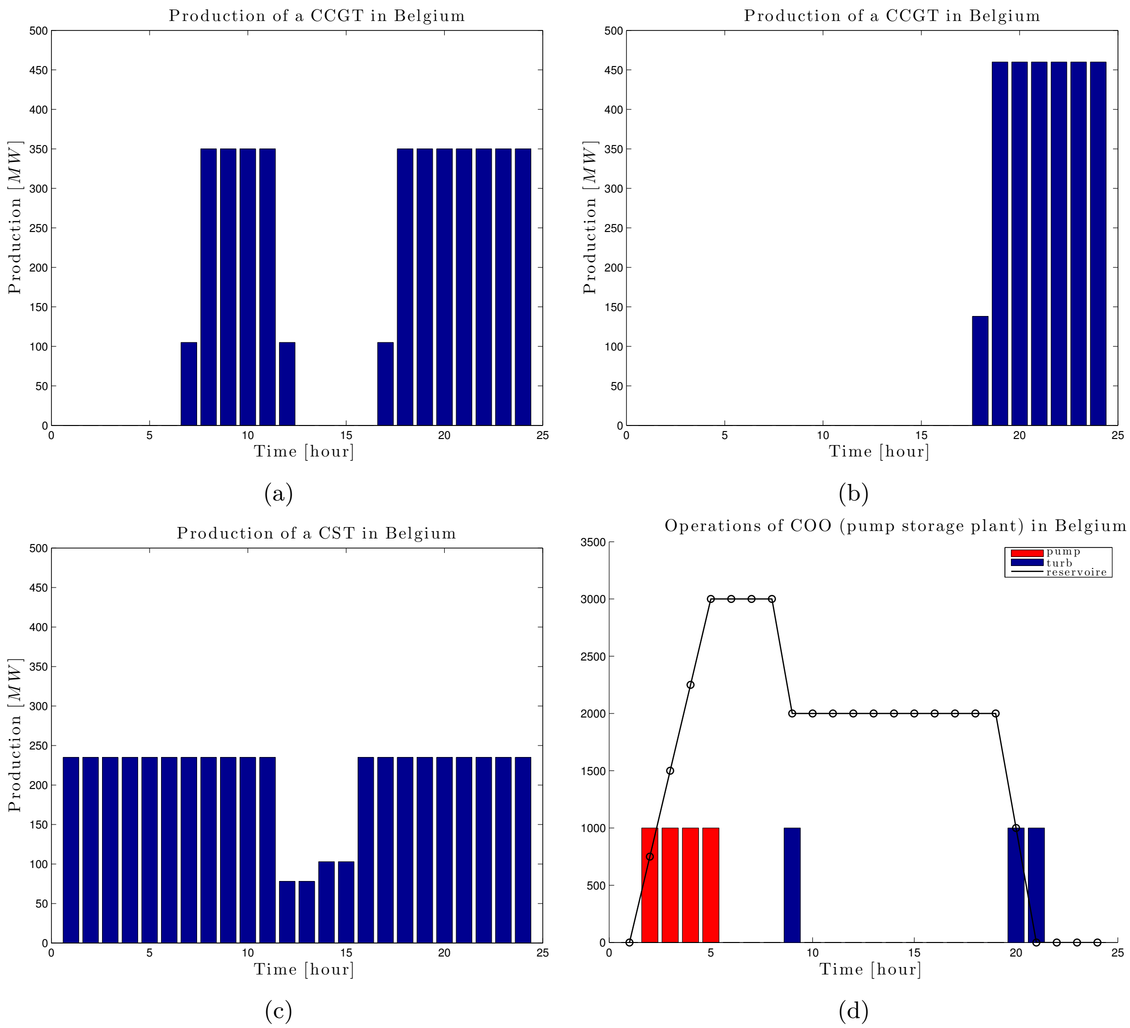


Figure 4.3: This figure illustrates the behaviour of two CCGT and one CST in Belgium facing price of figure 4.1a and demand of figure 4.1c. According to their cost, some non-convex units would like to stay on whereas other are switch on/off when the system needs it. It should be noted that even if only two particular CCGT units and one particular CST unit are shown here, the results are quite general : CST are often ON for long periods (in Gemrany, Belgium...) whereas CCGT are switch on/off for shorter periods, i.e. CCGT provide most of the flexibility in the system. The last figure shows the behaviour of coo pump-storage power plant in Belgium. It pumps water during the cheapest hours in the night and it releases water through turbines at the hours of highest prices in the day.

Algorithm	iteration number	CPU [s]	Lagrangian value
Level method	18	258	100439054.03

Table 4.2: Information of the run of the Level method solving the full case described at section 4.2 with 24 hours and a precision $\epsilon = 10^{-5}$. The algorithm ran on a Intel Core i5 processor using python multiprocessing on 3 cores. The corresponding results prices are showed figures 4.1 and 4.2.

4.3.2 Convergence behaviour

Chapter 3 analyses in detail the behaviour of the Level algorithm on several study cases. However, let's present a single result which demonstrates the efficiency of this algorithm (combine with the dualization of the convex parts as explained at section 3.6 and detailed at appendix A) on a bigger study case. Table 4.2 shows the computational statistics. It is obvious that it converges pretty fast : only 18 iterates for a CPU time of less than 5 minutes which makes such algorithm suitable for industrial applications (in trading for instance). The epsilon stopping criterion has been a bit lowered as a substantial part of the objective are now referring to network constraints or pump-storage profit. Even if the master has been complexified adding convex plants, pump-storage, network constraints... (see appendix A) it did not damage the computational performances. This is probably due to the fact that if such new features add complexity, it also adds constraints which prevent the prices from moving to much and "guide" him to the right solution... Figure 4.4 presents the price in Belgium iterate to iterate. It is interesting to see that the shape changes gently till the convex hull price and it is quickly pretty much acceptable.

Remark Let's notice an unexpected issue. The Level algorithm involves solving a quadratic program. This is done using state-of-the-art solvers such as `cplex` or `gurobi`. However, it appears that our problem is badly conditioned. This is due to the large range of coefficients in the matrix of constraints and the objective. Therefore, in some cases, the program collapses as the solver fails to find a solution. Usually it works for a 24 hours case. But with 72 hours, `gurobi` is usually unable to solve the projection program... Playing with the α parameter can help to deal with this unstable behaviour but it depends from case to case...

4.3.3 Reserve and market prices

In this section, a model **with** reserve is considered. The theoretical background explaining how reserve is included with convex hull pricing is explained at section 1.5. In this work, *ramp-up spinning reserve* is the only type of reserve considered. The reserve targets $R_{z,t}$ of each zones is defined as (the data are inspired from [1]) : $140[MW]$ for Belgium, $300[MW]$ for Netherlands, $2000[MW]$ for France and Germany, and $0[MW]$ for the other countries as there is not enough data about their generators to make it relevant...

In our model, the only units which are allowed to provide reserve are the non-convex plants (i.e. CCGT, CST, GT and CGT). This choice is driven by two observations : (1) these are the units providing the flexibility in the system and therefore it sounds natural to allow them to use reserve and (2) in practice, most of the time, aFRR (i.e. automatic frequency restoration reserve, this includes ramp-up spinning reserve) is provided in Belgium by CCGT units[4].

[4]According to the CREG.

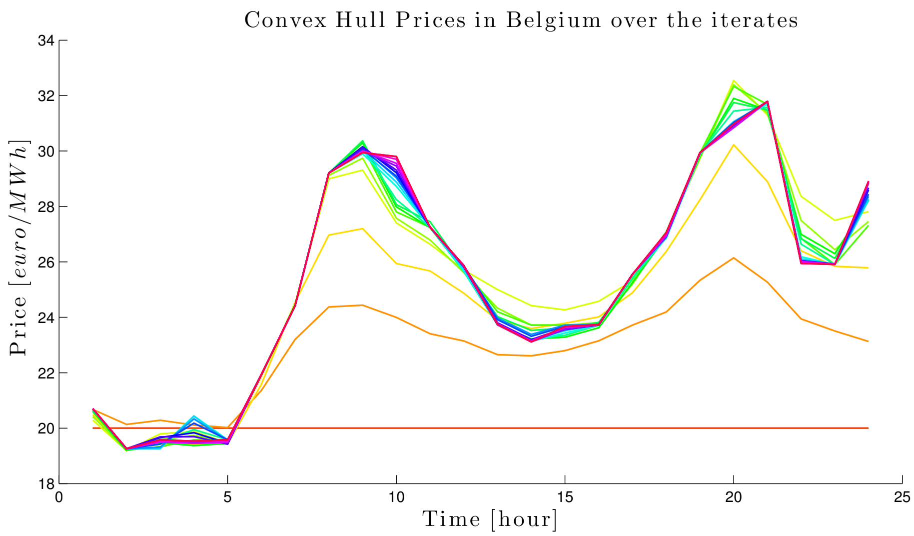


Figure 4.4: This figure illustrates the price vector for Belgium at each iterate of the algorithm. It starts with a price of 20 [€/MWh] for the 24 hours and it transforms progressively till the CHP. It demonstrates that the method behaves pretty well on such a huge study case and that the iterates converge fast till the convex hull price without making big moves and without oscillating.

In terms of computational effort, for the same study case, the model with 24 hours without reserve takes around 5 minutes whereas the model with reserve takes 15 minutes. A model with reserve for 72 hours takes around 30-40 minutes.

Reserve v.s. no reserve model Figure 4.5 illustrates the convex hull price obtained using a model with or without reserve. Clearly, there is not a big trend showing up from these prices... The reserve does not seem to influence the convex hull price of energy. For the rest of the tests of this chapter, we consider a model with reserve.

Convex hull price for the energy Let's make a simulation over three days (72 hours). Figure 4.6 displays the results obtained. Let's analyse the price of the energy (figure 4.6b).

- spikes in the convex hull prices or market prices correspond to periods of high demand.
- a general observation is that convex hull pricing provides a far better estimate of the real price than what the economic dispatch price does. Furthermore, CHP are always above ED prices.
- Comparing the real prices of the market and the convex hull price, it is clear that CHP achieves to locate the real price spikes. Nevertheless, the height of such spikes does not always fit with the real prices : the 4th and 5th price spikes does but the others under-estimate the height of the real price. However, putting the analysis in perspective with the demand (figure 4.6c), it does not seem clear why the real price spikes number 1, 2, 3 or 6 are actually higher than spikes 4 or 5. The elevation of the spikes produced by convex hull price seems consistent

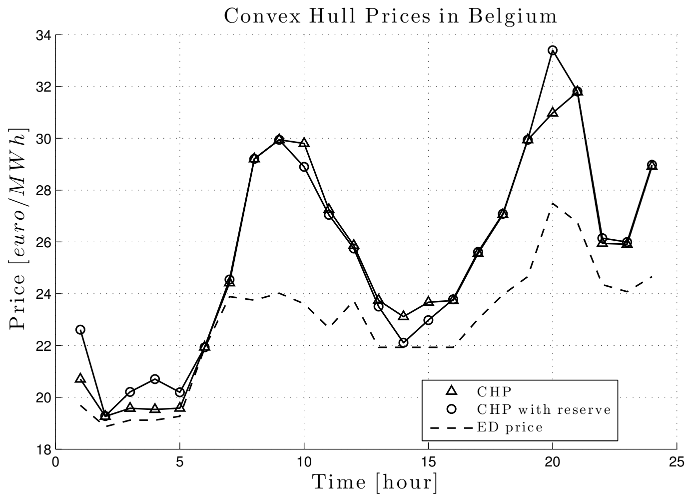


Figure 4.5: Convex hull price for 24 hours (14/03/2016) using full model of section 4.2. CHP denotes the model without reserve.

with the corresponding elevation of demand and it is not the case with real prices... the mismatch of elevation between CHP and market price is maybe due to some outages (which are not considered in the model) that happened that days and increased the real prices, or it is simply due to the lack of consistency of the real prices which can not be really explained by any fundamental pricing model such as CHP...

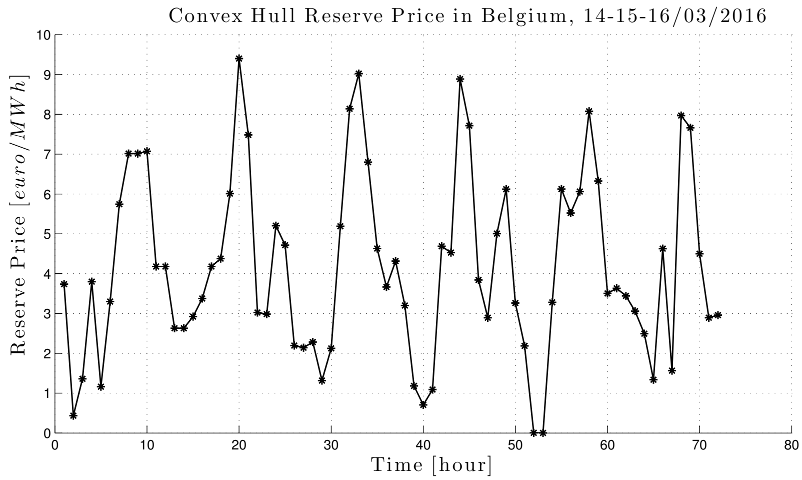
Reserve Price Theoretically, the reserve price should be equal to the opportunity cost of being out of the market. Indeed, let's imagine a unit A with a production cost of 40 [€/MWh] and a market price of 50 [€/MWh]. In this situation, A earns 10 [€/MWh] for each MW of produced power. Therefore, A would like to provide reserve if each MW of reserve is paid 10 [€/MWh] or more (i.e. remember that providing reserve does not cost anything).

Reserve price in our case is presented at figure 4.6a. Let's point out some observations

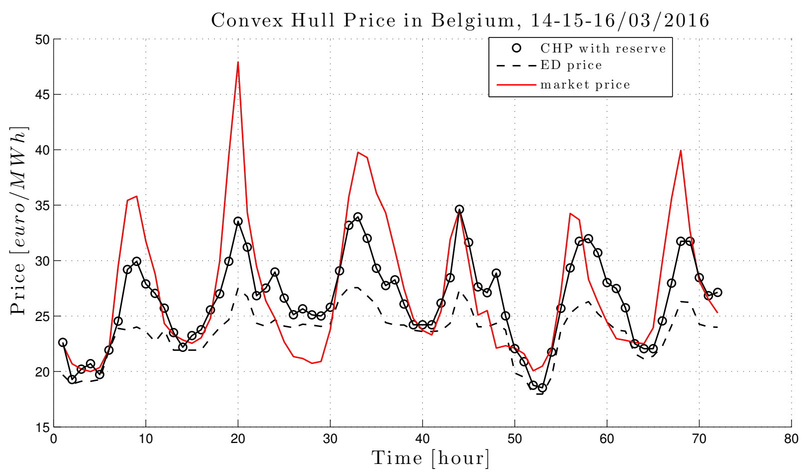
- High prices of reserve are generally located at periods with high demand and so periods with a high energy price.
- The previous reasoning explains the result. Indeed, CCGT, CST, GT and CGT units are the only ones allowed to provide reserve but at the same time they are also the units closed to the "at-the-money" (unlike nuclear for instance) so their opportunity cost of being out of the market is rather low and close to zero. Therefore, reserve price varies at a rather low level (below 10 [€/MWh] on the figure) and increases with the price of energy as a higher energy price means a higher opportunity cost of being out of the market.
- the average reserve price in Belgium was around 20-25 [€/MWh] in 2013 (see [1]). This is clearly higher then our result. A possible explanation for that difference is the following. In real market, most of the time, aFRR is provided by CCGT in Belgium. These units are often out of the money and at the same time Elia want to be sure that the amount of

aFRR required will be available. Therefore, in practise the reserve price can be either equal to their opportunity cost of being out of the market (if they are in-the-money; that is the case predicted by the theory) or equal to their recovery cost (if they are out-of-the-money but asked to provide reserve). This second option, which has no theoretical meaning but is practically intuitive can explain the difference between the price of figure 4.6a and the real rice of 20-25 [€/MWh].

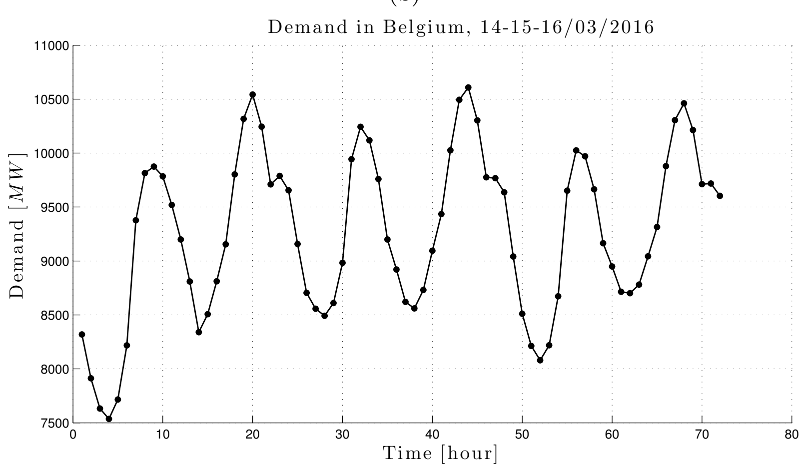
The reader, looking for additional results, similar to figure 4.6, in order to confirm the trends, can find them in the appendix B.



(a)



(b)



(c)

Figure 4.6: Results for 72 hours (14/03/2016 to 16/03/2016) in Belgium, implementing the full model descried at section 4.2 including reserve.

Conclusion and Perspectives

The introduction mentioned that this thesis had three goals : to establish the economical and mathematical models; to develop and efficient algorithm and to test it on the European model. Chapter 1 addressed the economical model, it established the reason why such elaborated pricing model was necessary and it briefly exposed its main properties. Chapter 2 tackled the question of an accurate and efficient mathematical representation of the unit commitment problem. It came up with a formulation which is a great deal between compactness and tightness. Chapter 3 debated the algorithmic question. Several algorithms presented in the literature have been compared and an innovative[5] algorithm, the so-called *Level method*, has been introduced and appeared to be highly efficient for solving this problem. The chapter also explained how more elaborated models, involving convex parts, can be incorporated with such algorithmic scheme. Finally, chapter 4 tested it on a European model with real data provided by ENGIE. This highlighted that Convex hull pricing was definitly better than marginal pricing in order to predict the market prices.

Main results

- Considering all the algorithms that have been implemented in this work, the Level method is clearly the most effective scheme to tackle the convex hull problem. It surpasses popular methods such as ACCPM. Figure 3.9 in chapter 3 gives a pretty clear illustration of this behaviour.

- Furthermore, the idea of dualize the convex parts in order to treat them directly in the master program, as it has been presented at section 3.6, is conclusive : it helps the Level method as it gives it a better "view" of the structure of the problem. Indeed, the tests of chapter 4 using more than 400 generators and including network constraints are solved quickly thanks to this dualization combine with an effective scheme such as the Level method.

- Regarding the use of this code for an industrial purpose, e.g. in trading departments, a 24 hours study case modelling the whole CWE market is solved by the algorithm in less than 15 minutes by a simple personal computer.

- As far as the results on a real study case are concerned, Convex Hull Price stands out as a highly promising fundamental pricing model. Indeed, figure 4.6 at chapter 4 have shown that CHP captures the real price variations and spikes in a very much accurate way than what marginal pricing would do.

[5]In the sense that it has never been applied on this specific convex hull pricing problem.

Improving paths

- As noted in chapter 4, the Level method needs to solve a quadratic program which for big systems may become numerically ill-conditioned. This is a concern as the pricing of electricity is a sensitive question which appeals robustness. Although this is a problem inherent to the solver itself (such as `gurobi`), it would perhaps be possible to improve the formulation of the problem in order to improve its numerical quality. Solving this issue would lead to a highly robust method as the Level algorithm itlself is robust as illustrated in chapter 3.

- The model that has been implemented in chapter 4 is already complex and includes many features. However, it might be interesting to includes additional features which could improve (even more!) the quality of the results.

  - In chapter 4, reserve was included into our model as it is an essential feature of current day-ahead markets. However, we only considered ramp-up spinning reserve. In practise, there are also ramp-down reserve, non-spinning reserve... Including them in the model could specify the impact of the reserve on the convex hull pricing and could maybe provide interesting results.
  - Without entering in too much details, it has been said in chapter 4 that some zones in the network were modelled using a flow-based approach. A well-known issue with such model is that it can produce *non-intuitive flows*[6]. In Europe, non-intuitive flows are not wished despite they are consistent with the fundamental pricing model. Therefore, an "artificial" trick has been triggered called *intuitive patch* which prevents such situations. As it is the way the auction proceeds, it would be interesting to include the intuitive patch into our model.

- Finally, if the whole thesis has been focused on convex hull price, it should be noted that other pricing schemes exist. In chapter 1, O'Neill pricing was mentioned as well as a pricing based on relaxing the integrity constraint. Another potentially promosing model is called *primal-dual* pricing as presented in [12]. An inherent and seemingly unavoidable concern with that scheme is that, unlike CHP which allows non-convex MIP programs to be decoupled, the primal-dual approach requires to solve big MIP programs which is a computational challenge. An interesting work could consist of comparing the former model with the CHP presented in this thesis.

[6] Let's consider a two-zones problem with zone z1 and z2. If the price of z1 is lower than the price of z2, it is economically intuitive that the flow will come from z1 to z2 (i.e. the zone with the higher price buys power to a zone with a lower price). A positive flow from z2 to z1 is called *non-intuitive* as it contradicts the economical intuition.

Bibliography

[1] Elia System Operator NV TenneT TSO BV TenneT TSO GmbH TransnetBW GmbH 50Hertz Transmission GmbH, Amprion GmbH. Potential cross-border balancing cooperation between the belgian, dutch and german electricity transmission system operators. 2014.

[2] Stephen Boyd, Lieven Vandenberghe, and Joëlle Skaf. Analytic center cutting-plane method, 2008.

[3] Pelin Damcı-Kurt, Simge Küçükyavuz, Deepak Rajan, and Alper Atamtürk. A polyhedral study of ramping in unit commitment. *Univ. California-Berkeley, Res. Rep. BCOL*, 13, 1777.

[4] C Gentile, G Morales-Espana, and A Ramos. A tight mip formulation of the unit commitment problem with start-up and shut-down constraints. *Institute for Research in Technology (IIT), Technical Report IIT-14-040A*, 2014.

[5] Paul R Gribik, William W Hogan, and Susan L Pope. Market-clearing electricity prices and energy uplift, 2007.

[6] Germán Morales-España, Claudio Gentile, and Andres Ramos. Tight mip formulations of the power-based unit commitment problem. *OR Spectrum*, 37(4):929–950, 2015.

[7] Germán Morales-España, Jesus M Latorre, and Angel Ramos. Tight and compact milp formulation for the thermal unit commitment problem. *Power Systems, IEEE Transactions on*, 28(4):4897–4908, 2013.

[8] Yurii Nesterov. *Introductory lectures on convex optimization: a basic course*. Springer, 2004.

[9] Anthony Papavasiliou. *Optimization Models and Algorithms in Electricity Markets*. Université Catholique de Louvain, 2015.

[10] Maurice Queyranne, Laurence Wolsey, et al. Tight mip formulations for bounded up/down times and interval-dependent start-ups. Technical report, UCL, 2015.

[11] Deepak Rajan and Samer Takriti. Minimum up/down polytopes of the unit commitment problem with start-up costs. *IBM Res. Rep*, 2005.

[12] Carlos Ruiz, Antonio J Conejo, and Steven A Gabriel. Pricing non-convexities in an electricity pool. *Power Systems, IEEE Transactions on*, 27(3):1334–1342, 2012.

[13] Dane A Schiro, Tongxin Zheng, Feng Zhao, and Eugene Litvinov. Convex hull pricing in electricity markets: Formulation, analysis, and implementation challenges. 2015.

[14] Andreas Schröder, Friedrich Kunz, Jan Meiss, Roman Mendelevitch, and Christian Von Hirschhausen. Current and prospective costs of electricity generation until 2050. *DIW Data Documentation*, 68:104, 2013.

[15] Matthias Silbernagl, Matthias Huber, and René Brandenberg. Improving accuracy and efficiency of start-up cost formulations in mip unit commitment by modeling power plant temperatures. 2014.

[16] Congcong Wang, Peter B Luh, Paul Gribik, Li Zhang, and Tengshun Peng. A subgradient-based cutting plane method to calculate convex hull market prices. In *Power & Energy Society General Meeting, 2009. PES'09. IEEE*, pages 1–7. IEEE, 2009.

[17] Gui Wang. *Design and operation of electricity markets: dynamics, uncertainty, pricing and competition*. PhD thesis, University of Illinois at Urbana-Champaign, 2013.

[18] Gui Wang, Uday V Shanbhag, Tongxin Zheng, Eugene Litvinov, and Sean Meyn. An extreme-point subdifferential method for convex hull pricing in energy and reserve markets—part i: Algorithm structure. *Power Systems, IEEE Transactions on*, 28(3):2111–2120, 2013.

[19] Gui Wang, Uday V Shanbhag, Tongxin Zheng, Eugene Litvinov, and Sean Meyn. An extreme-point subdifferential method for convex hull pricing in energy and reserve markets—part ii: Convergence analysis and numerical performance. *Power Systems, IEEE Transactions on*, 28(3):2121–2127, 2013.

Appendix A

Complete Convex Hull price model

Let's extend a bit the nomenclature. [1]

Sets	
$tp \in G^{hydro}$	Set of pump storage facilities
$g \in G^{conv}$	the subset of convex plants of G (i.e. the plants without start-up costs)
$g \in G^{bin}$	the subset of non-convex plants of G (i.e. the plants with start-up costs)
$z_f, z_a \in \mathcal{Z}$	Set of zones in the network, z_f being flow-based and z_a being ATC
$cb \in \mathcal{L}^{fb}$	the set of "flow-based" lines
$lk \in \mathcal{L}^{\mathcal{ATC}}$	the set of "ATC" lines
$\mathcal{T}^{season}$	a subset of $\mathcal{T}$ for which there is a target level for the pump-storage
Mappings	
$or_{lk,z}$	the subset of ATC lines lk coming from zone z
$dest_{lk,z}$	the subset of flow-based lines cb aiming zone z
Parameters	
$exo_{z,t}$	the exo-production in zone z at time t
tax^p	the tax for pumping
tax^t	the tax for realising water
η_g^{turb}	the efficiency of realising water
η_g^{pump}	the efficiency of pumping water
Vol_{tp}	the volume of the reservoir of tp
Vol_{tp}^{season}	the target volume of the reservoir for $t \in \mathcal{T}^{season}$
P_{tp}^{pump}	the maximal amount of water pumped
P_{tp}^{turb}	the maximal amount of water realised
$PTDF_{cb,z,t}$	Power transfer distribution factor (PTDF) of the network
$RAM_{cb,t}$	remaining available margin for flow-based line cb at time t (capacity of the line)
$F_{lk,t}$	capacity of the ATC line lk at time t

[1] Briefly on the lines sets, it is possible to write a model that takes into account the "physic" of the system, i.e. the Kirschkof laws which end up to the $PTDF$ constraints, this is roughly the flow-based approach. The ATC approach does not take into account the physical laws. In practise, some lines are flow-based and others are ATC.

Variables

$p^{turb}_{tp,t}$	released flow by the hydro plant tp
$p^{pump}_{tp,t}$	pumped flow by the hydro plant tp
$level_{tp,t}$	level of the reservoir of the hydro plant tp
$e_{z,t}$	Power shipped from node $z \in \mathcal{Z}$ to the "hub node" at time $t \in \mathcal{T}$
$f_{lk,t}$	the flow on line lk at time t

The scope of this appendix is to state the complete model as it has been implemented in GAMS and Python. The initial complete problem can be stated as follow, where $f_{g,t}$ denote the cost function of generator g for producing at time t.

$$
\begin{aligned}
&\min \sum_t \left(\sum_{g \in G^{conv}} f_{g,t} + \sum_{g \in G^{bin}} f_{g,t} + \sum_{tp \in G^{hydro}} tax^p p^{pump}_{tp,t} + tax^t p^{turb}_{tp,t} \right) \\
s.t.\ (\rho_{z,t})\quad & D_{z,t} + e_{z_f,t} - exo_{z,t} = \sum_{g \in G^{conv}} \widehat{p}_{g,t} + \sum_{g \in G^{bin}} p_{g,t} + P^{\min}_g u_{g,t} + \sum_{tp \in G^{hydro}} \eta^{turb}_{tp} p^{turb}_{tp,t} - p^{pump}_{tp,t} \\
&\quad + \sum_{lk \in dest(lk,z)} f_{lk,t} - \sum_{lk \in or(lk,z)} f_{lk,t} \\
& u_{g,t}, p_{g,t}, v_{g,t}, w_{g,t}, \delta_{g,s,t} \in X^{bin}_{g,t} \\
& \widehat{p}_{g,t} \in X^{conv}_{g,t} \\
& e_{z_f,t} \in X^{e}_{z,t} \\
& p^{turb}_{tp,t}, p^{pump}_{tp,t}, level_{tp,t} \in X^{hydro}_{tp,t} \\
& f_{lk,t} \in X^{flow-ATC}_{lk,t}
\end{aligned}
\tag{A.1}
$$

All this master thesis has been dedicated to the treatment of G^{bin}. The expressions of the equations in each set of constraints $X^{flow-ATC}_{lk,t}$... are detailed later on, but what is important to note here is that each set of variables corresponding to a different subset of constraints becomes independent as soon as the market clearing constraint is relaxed. After relaxation the model becomes

$$
\begin{aligned}
&\max_{\rho} \sum_{z} \sum_{t} \rho_{z,t} \left(D_{z,t} - exo_{z,t}\right) \\
&\quad + \min_{e \in X^{e}_{z,t}} \left\{ \sum_{z_f} \sum_{t} \rho_{z,t} e_{z_f,t} \right\} \\
&\quad + \min_{f \in X^{flow-ATC}_{lk,t}} \left\{ \sum_{t} \sum_{z} \rho_{z,t} \left(- \sum_{lk \in dest(lk,z)} f_{lk,t} + \sum_{lk \in or(lk,z)} f_{lk,t} \right) \right\} \\
&\quad - \max_{\substack{u,v,w,p,\delta \\ \in X^{bin}_{g,t}}} \left\{ \sum_{z} \sum_{t} \rho_{z,t} \sum_{g \in G^{bin}_{z}} \left(p_{g,t} + P^{\min}_{g} u_{g,t}\right) - \sum_{t} \sum_{g \in G^{bin}} f_{g,t} \right\} \\
&\quad - \max_{\widehat{p} \in X^{conv}_{g,t}} \left\{ \sum_{z} \sum_{t} \rho_{z,t} \sum_{g \in G^{conv}_{z}} \widehat{p}_{g,t} - \sum_{t} \sum_{g \in G^{conv}} f_{g,t} \right\} \\
&\quad - \max_{\substack{p^{turb}, p^{pump}, \\ level \in X^{hydro}_{tp,t}}} \left\{ \sum_{z} \sum_{t} \rho_{z,t} \sum_{\substack{tp \in \\ G^{hydro}_{z}}} \left(\eta^{turb}_{tp} p^{turb}_{tp,t} - p^{pump}_{tp,t}\right) - \sum_{\substack{tp \in \\ G^{hydro}}} \left(tax^{p} p^{pump}_{tp,t} + tax^{t} p^{turb}_{tp,t}\right) \right\}
\end{aligned}
\tag{A.2}
$$

Where the $-\max$ comes from $\min f = -\max -f$ and has been done as a "profit maximization" for the plant, convex or not, is economically more meaningful.

This Lagrangian problem has basically five components, respectively the network flow-based, the network ATC, the non-convex generators, the convex generators and the pump storage plants. The idea which has already been developed at section 3.6 is that some components are convex and therefore can be dualized and incorporated directly into the master program. The treatment of the convex and non-convex plants has already been stated explicitly in section 3.6. The others are detailed here after.

A.1 Dualization of the network flow-based equations

The network flow based sub-problem of (A.2) can be stated as follows

$$
\begin{aligned}
\min_{e} \quad & \sum_{t} \sum_{z_f} \rho_{z,t} e_{z_f,t} \\
s.t.\ (\theta_t) \quad & \sum_{z_f} e_{z_f,t} = 0 && \forall t \\
(\pi_{cb,t}) \quad & \sum_{z_f} PTDF_{cb,z,t} e_{z_f,t} \leq RAM_{cb,t} && \forall cb, t
\end{aligned}
\tag{A.3}
$$

And the dual is

$$
\begin{aligned}
\max_{\theta,\pi} \quad & -\sum_{t}\sum_{cb}\pi_{cb,t}RAM_{cb,t} \\
s.t. \quad & \rho_{z_f,t} + \theta_t + \sum_{cb}\pi_{cb,t}PTDF_{cb,z_f,t} = 0 \quad && \forall z_f, t \\
& \pi_{cb,t} \geq 0 && \forall cb, t
\end{aligned}
\tag{A.4}
$$

So if the PTDF constraint is non-binding, then $\pi_{cb,t} = 0$ and the price is the same in each zone.

A.2 Dualization of the network ATC equations

The ATC network sub-problem of (A.2) can be stated as follows

$$
\begin{aligned}
\min_{f \in X_{lk,t}^{flow-ATC}} \quad & \sum_{t}\sum_{z}\rho_{z,t}\left(-\sum_{lk \in dest(lk,z)} f_{lk,t} + \sum_{lk \in or(lk,z)} f_{lk,t}\right) \\
s.t. \quad (\xi_{lk,t}) \; & f_{lk,t} \leq F_{lk,t} && \forall lk, t \\
& f_{lk,t} \geq 0 && \forall lk, t
\end{aligned}
\tag{A.5}
$$

And the dual is

$$
\begin{aligned}
\max_{\xi} \quad & -\sum_{t}\sum_{lk}\xi_{lk,t}F_{lk,t} \\
s.t. \quad & -\sum_{dest(lk,z)}\rho_{z,t} + \sum_{or(lk,z)}\rho_{z,t} + \xi_{lk,t} \geq 0 \quad && \forall lk, t \\
& \xi_{lk,t} \geq 0 && \forall lk, t
\end{aligned}
\tag{A.6}
$$

meaning that the program tries to keep the price difference between two zones equal to 0 as long as possible (i.e. as long as the line is not congested). If the line is congested (primal constraint is binding) then the dual variable ξ is non zero and is equal to the price difference between the zones.

A.3 Dualization of the pump-storage equations

The pump-storage sub-problem of (A.2) can be stated as follows

$$
\begin{aligned}
- \max_{p^{turb}_{tp,t}, p^{pump}_{tp,t}, level_{tp,t}} \quad & \Bigg\{ \sum_z \sum_t \rho_{z,t} \sum_{g \in G^{hydro}_z} \eta^{turb}_g p^{turb}_{tp,t} - p^{pump}_{tp,t} \\
& - \sum_t \sum_{g \in G^{hydro}} tax^t p^{turb}_{tp,t} + tax^p p^{pump}_{tp,t} \Bigg\} \\
s.t. \; (\lambda_{tp,t}) \quad & level_{tp,t} = level_{tp,t-1} + \eta^{pump}_g p^{pump}_{tp,t} - p^{turb}_{tp,t} & \forall tp, t \geq 1 \\
(\lambda_{tp,0}) \quad & level_{tp,t} = Vol^{season}_{tp} + \eta^{pump}_g p^{pump}_{tp,t} - p^{turb}_{tp,t} & \forall tp, t = 0 \\
(\tau_{tp,t}) \quad & level_{tp,t} = Vol^{season}_{tp} & \forall tp, t \in \mathcal{T}^{season} \\
(\phi_{tp,t}) \quad & p^{turb}_{tp,t} \leq P^{turb}_{tp} & \forall tp, t \\
(\psi_{tp,t}) \quad & p^{pump}_{tp,t} \leq P^{pump}_{tp} & \forall tp, t \\
(\sigma_{tp,t}) \quad & level_{tp,t} \leq Vol_{tp} & \forall tp, t \\
& p^{turb}_{tp,t}, p^{pump}_{tp,t}, level_{tp,t} \geq 0 & \forall tp, t
\end{aligned}
\tag{A.7}
$$

And the dual is

$$
\begin{aligned}
- \min_{\lambda,\tau,\phi,\psi,\sigma} \quad & \Bigg\{ - \sum_{tp} \lambda_{tp,0} Vol^{season}_{tp} - \sum_{t \in \mathcal{T}^{season}} \sum_{tp} \tau_{tp,t} Vol^{season}_{tp} \\
& + \sum_t \sum_{tp} \phi_{tp,t} P^{turb}_{tp} + \sum_t \sum_{tp} \psi_{tp,t} P^{pump}_{tp} + \sum_t \sum_{tp} \sigma_{tp,t} Vol_{tp} \Bigg\} \\
s.t. \quad & - \rho_{t,z(tp)} \eta^{turb}_{tp} + tax^t - \lambda_{tp,t} + \phi_{tp,t} \geq 0 & \forall tp, t \\
& \rho_{t,z(tp)} + tax^p + \lambda_{tp,t} \eta^{pump}_{tp} + \psi_{tp,t} \geq 0 & \forall tp, t \\
& \sigma_{tp,t} + \lambda_{tp,t+1} (\textbf{if } h < T) - \lambda_{tp,t} - \tau_{tp,t} (\textbf{if } h \in \mathcal{T}^{season}) \geq 0 & \forall tp, t \\
& \psi_{tp,t}, \phi_{tp,t}, \sigma_{tp,t} \geq 0 & \forall tp, t
\end{aligned}
\tag{A.8}
$$

A.4 The complete model

Replacing the primal sub-problems by the duals (the convex plants at (3.19); ATC at (A.6); flow-based at (A.4) and pump storage at (A.8)) and using the "cuts" approximation for the non-convex generators such as developped in section 3.6, the complete "Master" problem of the Level method for finding the convex hull price is

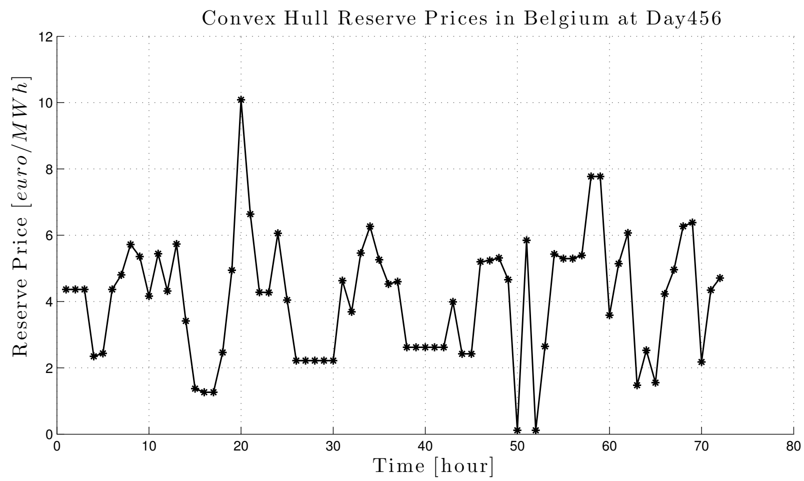
$$
\begin{aligned}
\max_{\rho,\pi,\theta\ldots} & \sum_{z}\sum_{t}\rho_{z,t}\left(D_{z,t}-exo_{z,t}\right) \\
& -\sum_{t}\sum_{cb}\pi_{cb,t}RAM_{cb,t} \\
& -\sum_{t}\sum_{lk}\xi_{lk,t}F_{lk,t} \\
& -\Bigg\{-\sum_{tp}\lambda_{tp,0}Vol_{tp}^{season}-\sum_{t\in\mathcal{T}^{season}}\sum_{tp}\tau_{tp,t}Vol_{tp}^{season} \\
& +\sum_{t}\sum_{tp}\phi_{tp,t}P_{tp}^{turb}+\sum_{t}\sum_{tp}\psi_{tp,t}P_{tp}^{pump}+\sum_{t}\sum_{tp}\sigma_{tp,t}Vol_{tp}\Bigg\} \\
& -\sum_{g\in G^{conv}}\sum_{t}\nu_{g,t}P_{g}^{\max}-\mu_{g,t}P_{g}^{\min} \\
& -profit^{bin} \\
s.t.\quad & \text{constraints (A.4), (A.6), (A.8), (3.19)} \\
& profit^{bin}\geq\langle a_i,\rho\rangle+b_i \qquad \forall i
\end{aligned}
\tag{A.9}
$$

where a_i and b_i are cuts coefficients such as detailed at section 3.6.

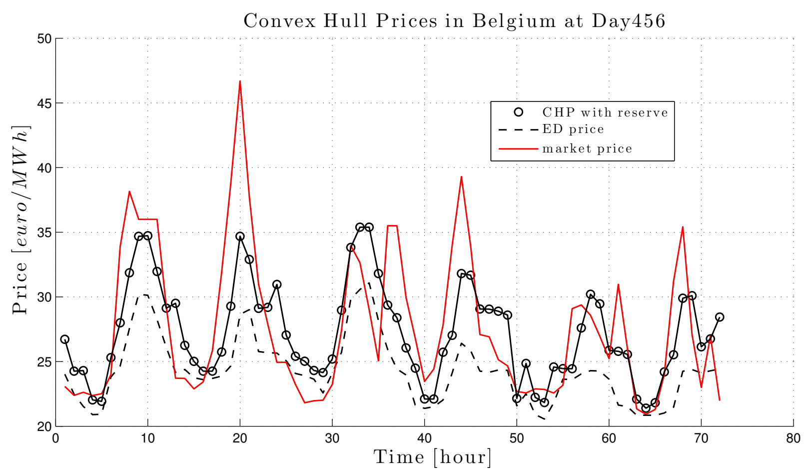
Appendix B

Additional results

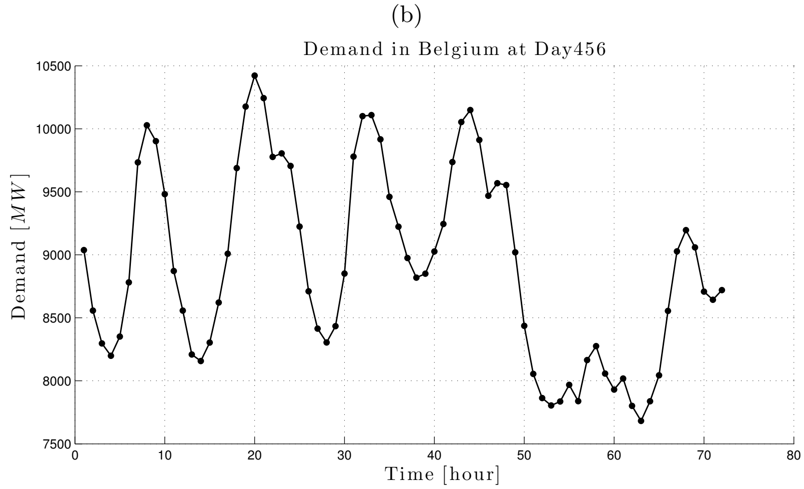
Figure 4.6 displays some results over three days in Belgium comparing CHP with ED prices and the real market prices. This appendix presents some additional results similar from the ones of chapter 4. The model used is exactly the same as the one used for figure 4.6 but on other days in March 2016. The results are shown at figure B.1, B.2 and B.3.



(a)

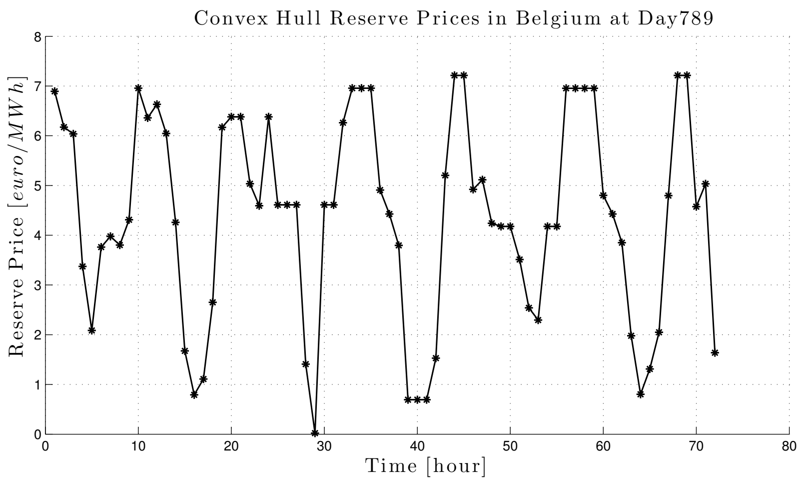


(b)

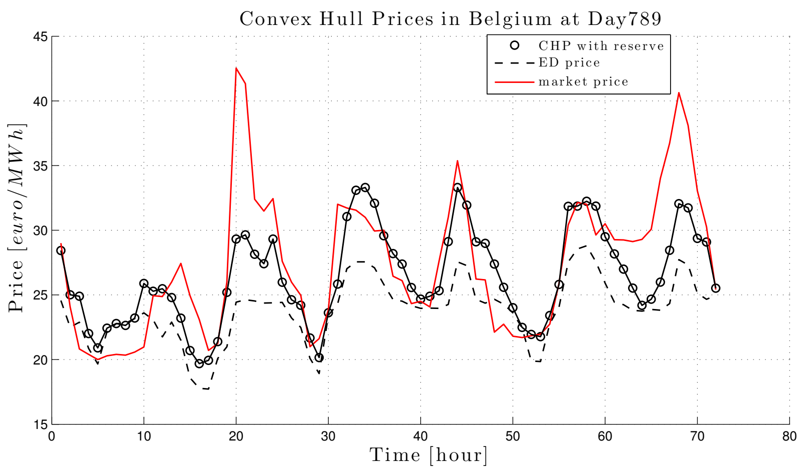


(c)

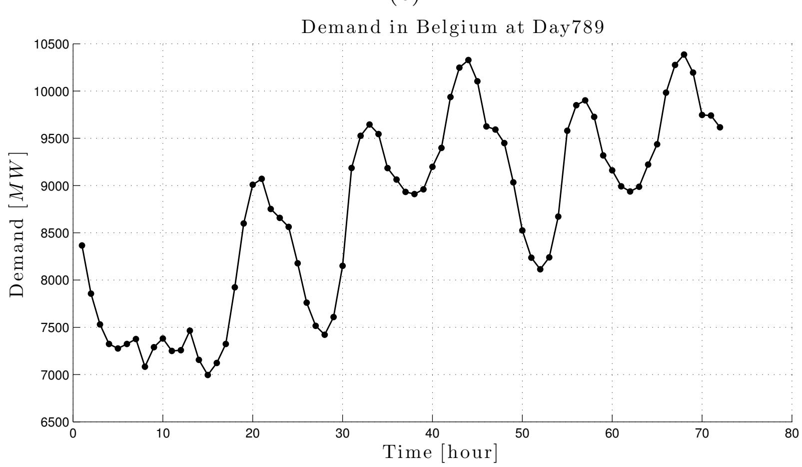
Figure B.1: Results for 72 hours (17/03/2016 to 19/03/2016) in Belgium, implementing the full model descried at section 4.2 including reserve.



(a)

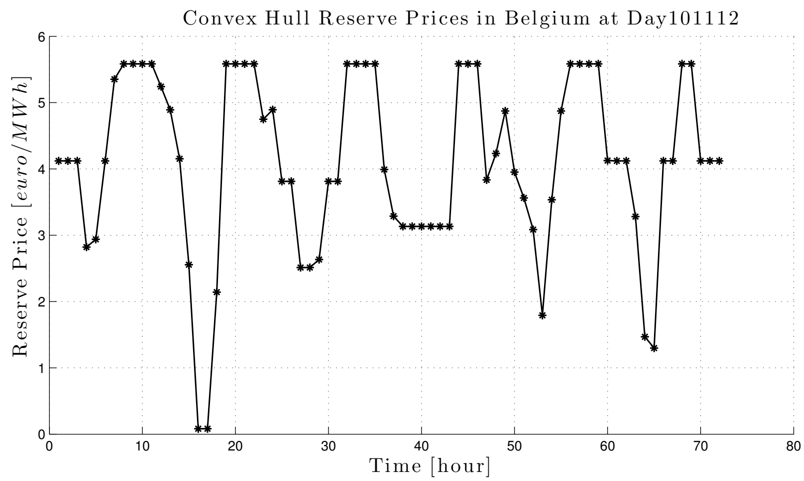


(b)

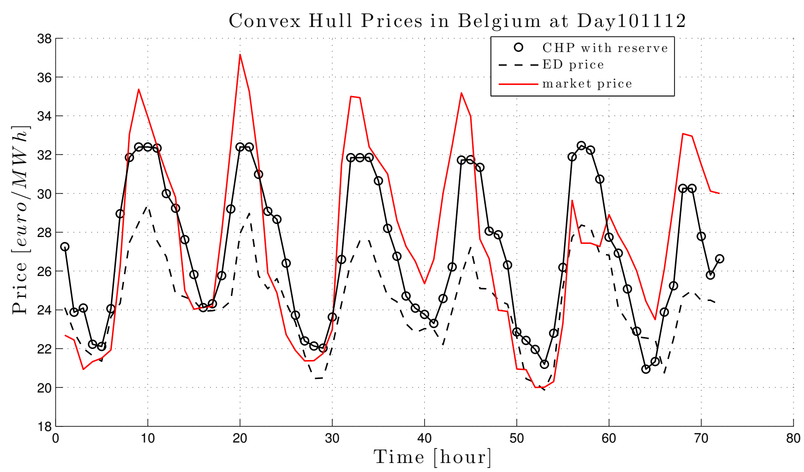


(c)

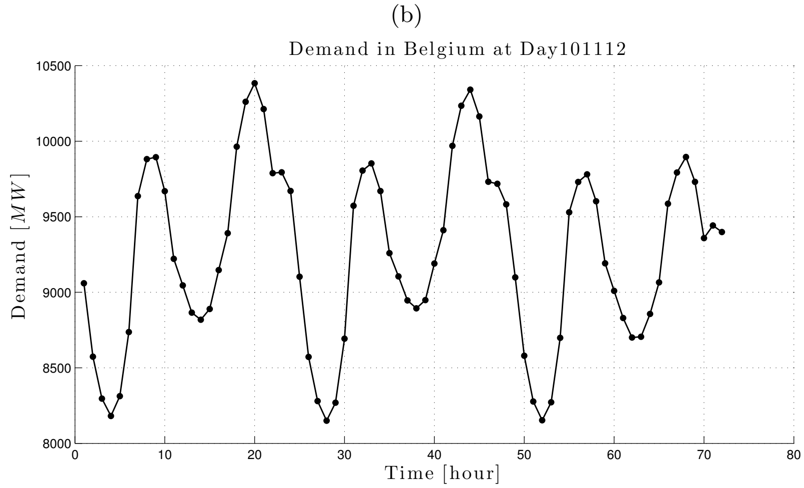
Figure B.2: Results for 72 hours (20/03/2016 to 22/03/2016) in Belgium, implementing the full model descried at section 4.2 including reserve.



(a)



(b)



(c)

Figure B.3: Results for 72 hours (23/03/2016 to 25/03/2016) in Belgium, implementing the full model descried at section 4.2 including reserve.

Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve www.uclouvain.be/epl