UCL Université catholique de Louvain

École polytechnique de Louvain (EPL)

epl

Optimal power dispatch

Analysis of a new method with global convergence

Dissertation presented by
Pierre-Alexandre LOUIS

for obtaining the Master's degree in
Mathematical Engineering

Supervisor(s)
Pierre-Antoine ABSIL

Reader(s)
Anthony PAPAVASILIOU, François GLINEUR

Academic year 2017-2018

Acknowledgments

First of all, I would like to thank my supervisor Pierre-Antoine Absil for proposing this topic,for supervising my research during this year and its availability to answer questions.

I would also like to thank François Glineur and Anthony Papavasiliou for agreeing to take the time to read my thesis.

Finally, I want to thank my family, specially my parents, for their support and their encouragement. They provided me with everything I needed without asking anything in return.

Contents

Introduction

The economic load dispatch is a common problem faced daily by power grid operators. The task is to find the optimum repartition of a system load among different electricity generation facilities or said differently to determine the lowest possible production cost subject to transmission and operational constraints. The general idea is to use the generators in function of their marginal cost. The unit whose marginal cost is the lowest will be employed first. Then, the one whose cost is the second lowest will be used and it continues recursively until the sum of the generated electricity reaches the demand.

A special constraint for the production is a source of specific complications. During the production, large units do not generate power smoothly. Instead, large steam turbines are controlled by several steam admission valves which operate independently. When a valve is open. The cost increase because of the throttling around the valves and because of the sudden temperature increase driven by the addition of fuel. This creates ripple curves over the cost function. This phenomenon is called the valve-point effect. Hence, the curb is no longer convex and classic methods of optimization cannot be employed.

In the literature, most of the approaches to solve this issue are based on heuristics. So, they cannot guarantee convergence to the optimal solution; only to a local minimum. The stopping point of the algorithm depends on the initialization of the algorithm. Henceforth, some procedures perform very well on some data whereas in order cases they do not reach the same quality.

In [1], a new model targeting this issue is proposed by minimizing a mixed-integer quadratic substitution problem which converges to a global optimum. The surrogate problem is known to return a cost lower than the real one. However, given a finite iteration number, the difference between the two converges to zero: there is no gap between the surrogate problem and the real one. Consequently, the global optimum is reached with absolute certainty.

The optimization of mixed-integer problem is known to use a lot of variables, which gives a low speed of convergence on large problems. Hence, the presented method does not intend to replace heuristic algorithms aiming to propose a compromise between accuracy and run time. Its purpose is to find the unknown optimality bound of different types of economic load dispatch problem.

The method, named Adaptive Piecewise-Quadratic Under-Approximation (APQUA) [1], has not already been tested on more complex models subject to other constraints than the valve-point effect. This document aims to apply the algorithm on other cases and to compare it with other approaches to get more knowledge on its performances. It is made on cases proposed in review papers in order to easily compare the outputs, with heuristic approaches for example.

In the first chapter of this thesis, some explanations about the theoretical background of the mixed-integer formulation are given. The model developed by the UCL team is described and afterwards, it is tested on a dataset to obtain some numerical results. This data is one of the reference test sets used in the field to compare the different algorithms.

The second chapter talks about the dynamic economic load dispatch problem. Here, the electricity generation is considered over several periods, which are generally hours. The production between two consecutive hours cannot change too widely since units need time to reach their maximum capacity or to cool down. APQUA is then tested and the consequences of a dispatch

over several hours are observed, in particular on the run time.

In the third chapter, new constraints are introduced for the static case. Generators are usually not restricted to only one usable fuel. Each combustible has its own properties and then its specific marginal cost functions. The end of the chapter compares the APQUA technique to other ones.

Finally, the fourth chapter talks about the loss of power during the production when some part is lost due to bad insulation of the network or due to heat associated with electricity generation. The losses are directly correlated to the production and they increase as the production does. Hence, even when disregarding the valve-point effect, the problem is no longer convex. Methods are proposed to avoid this and are then numerically tested.

Chapter 1

Context and original model

The basics of the problem of the economic load dispatch problem (ELDP) and the proposed solution given by the team of the Université Catholique de Louvain [1], on which the present master thesis in based, are detailed in the present chapter. Then, the performance of the algorithm is tested on a reference dataset of the field.

1.1 The economic dispatch problem

The mathematical context around the economic dispatch problem with valve-point effect will be detailed in this section. This is an optimization problem. The cost function to produce electricity thanks to several reactors has to be minimized. Each reactor produces a certain amount of electricity which impacts on the objective function. It can be written as:

$$f(p) = \sum_{i=1}^{n} f_i(p_i) \tag{1.1}$$

where f is the total production cost, n is the number of reactors, f_i is the individual electric production cost of each reactor. It is a function of p_i, the electricity produced by reactor i.

A classical way to model this cost function f_i is a quadratic function for each generator, as follows,

$$f_i(p_i) = a_i p_i^2 + b_i p_i + c_i$$

where a_i,b_i and c_i are scalar coefficients.

However, actually, performance curves doe not behave smoothly. Instead, a generator with multi-valve turbines shows a variation in the cost function. The valve opening process of the reactors induces ripples on those curves. The wire drawing effect leads to a loss when the steam admission valve starts to open [20], [21]. This phenomenon is called the valve-point effect (VPE). A regular model for this effect is the rectified sine curve:

$$f_i^{VPE}(p_i) = d_i |sin(e_i(p_i - p_i^{min}))| \tag{1.2}$$

where d_i and e_i are both positive coefficients [23]. The cost function is:

$$f_i(p_i) = a_i p_i^2 + b_i p_i + c_i + d_i |sin(e_i(p_i - p_i^{min}))|$$

A visualisation of the cost function with and without valve-point effect is shown in the figure 1.1.

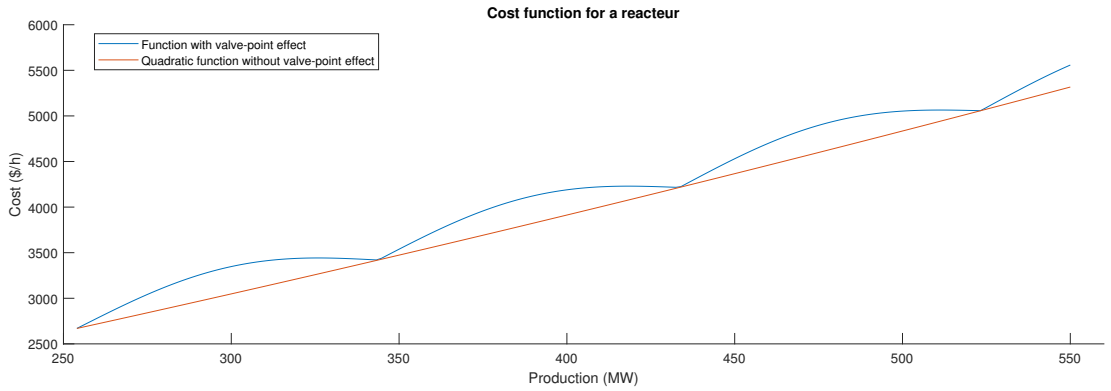


Figure 1.1: Production cost for a reactor. The valve-point effect creates non smooth curves.

In [1], the VPE function f_i^{VPE} (1.2) is only considered for each i as a Lipschitz-continuous piecewise-concave function. Further information on the requirements will be provided in section 1.2.1.

Some constraints also have to be satisfied to correspond to the ELDP. First, a constraint corresponding to the power balance constraint. It imposes that the sum of the production of each reactor is equal to the demand even though some power is wasted in the production:

$$\sum_{i=1}^{n} p_i = D + p_L(p)$$

with D a scalar and $p_L(p)$ representing respectively the total power demand and the power loss during the transmission over the network in MW. In this basic version of the problem, the power-loss term is set to 0; in the section 4, consequences of a non-null coefficient are explored.

Another constraint, the physical production limitations of the reactors, has to be applied; each of them has a minimum,p_i^{min} and a maximum,p_i^{max}. The power generation is within the interval $[p_i^{min}, p_i^{max}]$. The restriction on the domain is written as:

$$p_i^{min} \leq p_i \leq p_i^{max} \ .$$

In the end, the ELDP without losses under the two above mentioned constraints and stated as a minimization problem is :

$$\underset{p \in \mathrm{R}^n}{\text{minimize}} \qquad f(p) = \sum_{i=1}^{n} a_i p_i^2 + b_i p_i + c_i + d_i |sin(e_i(p_i - p_i^{min}))| \tag{1.3a}$$

$$\text{subject to} \qquad \sum_{i=1}^{n} p_i = D \tag{1.3b}$$

$$p_i^{min} \leq p_i \leq p_i^{max} \quad \forall i \in \{1, \ldots, n\} \ . \tag{1.3c}$$

1.2 Solving way

The approach proposed by the UCL team will be shown and the algorithm explained in this section.

1.2.1 Surrogate problem

The problem posed in the previous section can be tackled by solving several surrogate problems in a row. Each of those is solved thanks to a mixed integer quadratic programming (MIQP) solver.

The surrogate problem consists in replacing f_i^{VPE} by its spline interpolation. Regarding the function, it is concave on the interval $[p_{i,j}^{kink}, p_{i,j+1}^{kink}]$,$j = 1, ...n_i^{kink} - 1$, with $p_{i,1}^{kink} \leq p_i^{min}$ and $p_{i,j+1}^{kink} \geq p_i^{max}$. The method used in [1] does not require a Lipschitz constant but needs to know the kink points. For the common rectified sine model (1.2), the kink points are given by:

$$p_{i,j}^{kink} = (j-1)\frac{\pi}{e_i} + p_i^{min} \qquad\qquad n_i^{kink} = \left\lceil (p_i^{max} - p_i^{min})\frac{e_i}{\pi} \right\rceil + 1 \ .$$

For every i, given knot $X_{i,1} < X_{i,2} < ... < X_{i,n_i^{knot}}$ including the kink points mentioned above, the spline interpolation with respect to the given knots is:

$$s_i(p_i) = \alpha_{i,j} p_i + \beta_{i,j} \ , p_i \in [X_{i,j}, X_{i,j+1}], \ j = 1, 2, \ldots, n_i^{knot} - 1 \tag{1.4}$$

where

$$\alpha_{i,j} = \frac{f_i^{VPE}(X_{i,j+1}) - f_i^{VPE}(X_{i,j})}{X_{i,j+1} - X_{i,j}}$$

is the slope of the jth linear part and

$$\begin{aligned} \beta_{i,j} &= f_i^{VPE}(X_{i,j}) - \alpha_{i,j} X_{i,j} \\ &= \frac{X_{i,j+1} f_i^{VPE}(X_{i,j+1}) - X_{i,j} f_i^{VPE}(X_{i,j})}{X_{i,j+1} - X_{i,j}} \end{aligned}$$

is the vertical intercept. Hence, the surrogate problem given by the knots $X_{i,1} < X_{i,2} < ... < X_{i,n_i^{knot}}$ is:

$$\underset{p \in \mathrm{R}^n}{\text{minimize}} \qquad g(p) = \sum_{i=1}^{n} a_i p_i^2 + b_i p_i + c_i + s_i(p_i) \tag{1.5a}$$

$$\text{subject to} \qquad \sum_{i=1}^{n} p_i = D \tag{1.5b}$$

$$p_i^{min} \leq p_i \leq p_i^{max} \ \ \forall i \in \{1, \ldots, n\} \ . \tag{1.5c}$$

The linear spline s_i is an underapproximation of the VPE term f_i^{VPE} according to the assumption that the function is concave. So, the function in 1.5 returns an underestimated cost of the original objective function 1.3a.

1.2.2 MIP theory

The chosen mixed-integer programming (MIP) formulation for the optimization of a piecewise-linear cost function

$$s(p) = \alpha_j p + \beta_j, \quad p \in [X_{i,j-1}, X_{i,j}], \ j = 1, 2, \ldots, n^{knot-1}$$

admits the special order set of type 1 (SOS1) formulation, termed multiple choice model in [9]:

$$\underset{p, \chi, \eta}{\text{minimize}} \qquad f(p) = \sum_{j=1}^{n^{knot}-1} \alpha_j \chi_j + \beta_j \tag{1.6a}$$

$$\text{subject to} \qquad p = \sum_{j=1}^{n^{knot}-1} \chi_j \tag{1.6b}$$

$$X_j \eta_j \leq \chi_j \leq X_{j+1} \eta_j \quad \forall j \in 1, \ldots, n^{knot} - 1 \tag{1.6c}$$

$$\sum_{j=1}^{n^{knot}-1} \eta_j = 1 \tag{1.6d}$$

$$\eta_j \in \{0, \ldots, 1\} \ \forall j \in 1, \ldots, n^{knot} - 1 \ . \tag{1.6e}$$

The constraint (1.6d) sets every η_j to zero except only one called $\hat{j}$ which equals 1. So, it forces the solver to select only one of the piecewise linear pieces. The constraint (1.6c) imposes $\xi_{\hat{j}}$ to belong to the interval $[X_{\hat{j}}, X_{\hat{j}+1}]$, every other one is zero. Hence, the constraint (1.6b) becomes $p = \chi_{\hat{j}}$; the objective function is also simplified to : $f(p) = \alpha_{\hat{j}}\chi_{\hat{j}} + \beta_{\hat{j}}$.

1.2.3 MIQP formulation of the surrogate problem

By executing the same method as above, the surrogate problem given in section 1.5 can rewriten as an MIQP problem. This model is a simpler version of the model in [26, §3.1]

$$\underset{p,\chi,\eta}{\text{minimize}} \quad g(p) = \sum_{i=1}^{n}\left[a_i p_i^2 + b_i p_i + c_i + \sum_{j=1}^{n_i^{knot}-1}(\alpha_{ij}\chi_{ij} + \beta_{ij}\eta_{ij})\right] \tag{1.7a}$$

$$\text{subject to} \quad \sum_{i=1}^{n} p_i = D \tag{1.7b}$$

$$p_i^{min} \leq p_i \leq p_i^{max} \tag{1.7c}$$

$$p_i = \sum_{j=1}^{n_i^{knot}-1} \chi_{ij} \tag{1.7d}$$

$$X_{i,j}\eta_{i,j} \leq \chi_{i,j} \leq X_{i,j+1}\eta_{i,j} \tag{1.7e}$$

$$\sum_{j=1}^{n_i^{knot}-1} \eta_{ij} = 1 \tag{1.7f}$$

$$\eta_j \in \{0, \ldots, 1\} \ . \tag{1.7g}$$

1.2.4 Knots update mechanism and algorithm explanation

The algorithm proposed in [1] and named Adaptive Piecewise-Quadratic Under-Approximation (APQUA) solves this problem. Furthermore, as shown in [1, §IV] , the objective function is guaranteed to converge to the global optimum. The algorithm alternates between obtaining, with an MIQP solver, a feasible approximate solution $\tilde{p}$ of the surrogate problem (1.5a) caused by the current set of knots, and updating the set of knots by inserting $\tilde{p}$ inside it.

In the implementation of APQUA, the initial knots (line 3 of Algorithm 1) are the kink points and, sometimes, their mid-points, which presence sometimes speeds up the algorithm.

The MIQP solver is called in line 12. The "if" statement in line 15 prevents repeated knots. The "sort" operation in line 16 is straightforward since it merely consists in inserting $\tilde{p}_i$ in the already sorted list. In line 13, g refers to the surrogate function (1.5a) induced by the current knots $X_{i,1}, \ldots, X_{i,n_i^{\text{knot}}}$.

Algorithm 1 APQUA: Adaptive piecewise-quadratic under-approximation

1: Set tolerance parameter $\delta_{\text{tol}} \geq 0$;
2: **for** $i = 1, \ldots, n$ **do**
3: Choose ordered knots $(X_{i,1}, \ldots, X_{i,n_i^{\text{knot}}})$ including the kink points $p_{i,1}^{\text{kink}}, \ldots, p_{i,n_i^{\text{kink}}}^{\text{kink}}$.
4: **end for**
5: **repeat**
6: **for** $i = 1, \ldots, n$ **do**
7: **for** $j = 1, \ldots, n_i^{\text{knot}} - 1$ **do**
8: $\alpha_{i,j} \leftarrow (f^{\text{VPE}}(X_{i,j+1}) - f^{\text{VPE}}(X_{i,j}))/(X_{i,j+1} - X_{i,j})$;
9: $\beta_{i,j} \leftarrow f^{\text{VPE}}(X_{i,j}) - \alpha_{i,j} X_{i,j}$;
10: **end for**
11: **end for**
12: $\tilde{p} \leftarrow$ feasible (approximate) solution of the MIQP surrogate problem (1.5), obtained with an MIQP solver with absolute optimality gap tolerance γ;
13: $\delta \leftarrow f(\tilde{p}) - g(\tilde{p})$;
14: **for** $i = 1, \ldots, n$ **do**
15: **if** $\min_{j \in \{1, \ldots, n_i^{\text{knot}}\}} |\tilde{p}_i - X_{i,j}| > 0$ **then**
16: $(X_{i,1}, \ldots, X_{i,n_i^{\text{knot}}+1})$
$\leftarrow \text{sort}(X_{i,1}, \ldots, X_{i,n_i^{\text{knot}}}, \tilde{p}_i)$
17: $n_i^{\text{knot}} \leftarrow n_i^{\text{knot}} + 1$;
18: **end if**
19: **end for**
20: **until** $\delta < \delta_{\text{tol}}$

An iteration of the algortihm is presented in the figure 1.2.

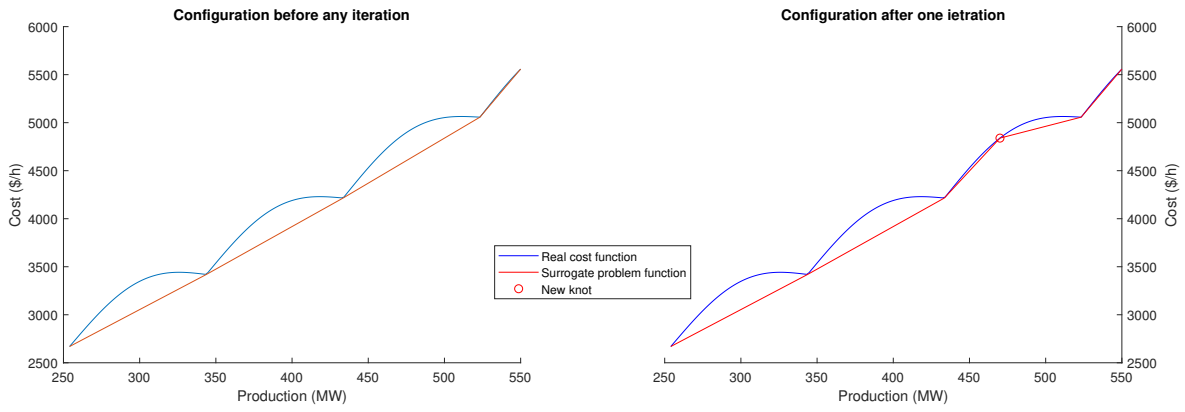


Figure 1.2: Introduction of a new points in the set of knots of a reactor after an iteration of the algorithm. It is noted that the gap between the linear spline and the cost function is small at the end of the plot before the iteration.

1.3 Hot start model

To speed up the convergence of the algorithm, a hot start can be tried: the problem is solved in a simpler version through a preliminary step. Then, the optimum points returned are the starting points of the first true iteration [25].

For the ELD problem with valve-point effect, the preliminary step consists of the same problem without VPE function. The rest remains the same. The line below is added between the lines 4 and 5 of algorithm:

$\tilde{p} \leftarrow$ feasible solution of the problem without valve-point effect.

1.4 Numerical results

In the ELDP literature, test sets may change widely from paper to paper. It is then requested to use reference set in order to validate the algorithm. In [11], the authors proposed a review of a large number of different methods as well as several reactor sets whose true optimum is known. This assures an objective comparison between methods. It also prevents heuristic methods from obtaining remarkable results on sets proposed by their authors.

APQUA is tested on the reference set proposed in [11]. The results for the ELDP with valve-point effect with single type of fuel and no power loss term are detailed in the table 1.1 and these are based on the data set from [22]. The total cost returned by the present method is 121,412.535509$/h, this cost is exactly equal, excluding approximation errors, to the one proposed in the review paper (121,412.5355$/h). This proves that the method does not lead to any mistake in the cost function since the lower bound is not crossed.

The algorithm with a hot start returns the same cost, as expected. However, its run time is larger than the version without hot start as shown in the table 1.2. The addition of the new iteration to the run time does not compensate the gain of starting at the outputs returned by the problem without VPE. In the following sections of this work, the optimizatoin problems are solved without hot start.

Unit	**Power (MW)**	**Unit**	**Power (MW)**
p_1	110.799825	p_{21}	523.279370
p_2	110.799825	p_{22}	523.279370
p_3	97.399913	p_{23}	523.279370
p_4	179.733100	p_{24}	523.279370
p_5	87.799905	p_{25}	523.279370
p_6	140.000000	p_{26}	523.279370
p_7	259.599650	p_{27}	10.000000
p_8	284.599650	p_{28}	10.000000
p_9	284.599650	p_{29}	10.000000
p_{10}	130.000000	p_{30}	87.799905
p_{11}	94.000000	p_{31}	190.000000
p_{12}	94.000000	p_{32}	190.000000
p_{13}	214.759790	p_{33}	190.000000
p_{14}	394.279370	p_{34}	164.799825
p_{15}	394.279370	p_{35}	200.000000
p_{16}	394.279370	p_{36}	194.397778
p_{17}	489.279370	p_{37}	110.000000
p_{18}	489.279370	p_{38}	110.000000
p_{19}	511.279370	p_{39}	110.000000
p_{20}	511.279370	p_{40}	511.279370
Total cost ($/h)	121412.535519		

Table 1.1: Total cost of production for the 40 reactors dataset.

	Regular model	Hot start
Run time (s)	2.42	2.86

Table 1.2: Run time for APQUA on the 40 reactors dataset with and without hot start.

Chapter 2

Dynamic Economic Dispatch Problem

This chapter details more complicated models dealing with several periods of time and other new constraints. The numerical results are shown at the end of the part.

2.1 Dynamic model with ramp-limit, spinning reserve constraints

The dynamic model is explained in this section. The economic load dispatch problem also applies during multiple periods of time. This introduces new constraints because continuity between time periods has to be assured,. Each reactor produces a given amount of electricity per hour. The total operation costs over the entire period have to be minimized.

The cost function is:

$$f(p) = \sum_{i=1}^{n} \sum_{t=1}^{m} f_{i,t}(p_{i,t})$$

where $p_{i,t}$ is the production of the reactor i at the period t and $f_{i,t}$ is the production cost for a given reactor i and for a certain time t, m is the number of hours.

Over the whole time interval, the cost function of the ELDP with valve-point effect is set to get:

$$f(p) = \sum_{t=1}^{m} \sum_{i=1}^{n} a_i p_{i,t}^2 + b_i p_{i,t} + c_i + d_i |sin(e_i(p_{i,t} - p_i^{min}))|$$

where the coefficients a_i, b_i, c_i, d_i, e_i are the ones of the static model.

Other constraints present in (1.3) also have to be adapted. The model is now:

$$\underset{p \in \mathrm{R}^n}{\text{minimize}} \quad f(p) = \sum_{t=1}^{m} \sum_{i=1}^{n} a_i p_{i,t}^2 + b_i p_{i,t} + c_i + d_i |sin(e_i(p_{i,t} - p_i^{min}))| \tag{2.1a}$$

$$\text{subject to} \quad \sum_{i=1}^{n} p_{i,t} = D_t \quad \forall t \in \{1, \ldots, m\} \tag{2.1b}$$

$$p_i^{min} \leq p_{i,t} \leq p_i^{max} \quad \forall i, t \in \{1, \ldots, n\}, \{1, \ldots, m\} \ . \tag{2.1c}$$

where D_t is the demand for the time interval t.

However, this model does not prevent discontinuity between periods. Indeed, the production of a reactor could be widely different between two consecutive hours; this is not physically possible because of the ramp limits. Generators needs time to either cool down or to heat up. The unit production from one period to the next cannot change too rapidly, either in the upward or

downward direction [27]. Hence a limit is imposed on that variation. This behaviour is prevented by the constraints:

$$-DR_i \leq p_{i,t} - p_{i,t-1} \leq UR_i \quad \forall i, t \in \{1, \ldots, N\}, \{2, ...T\}$$

where DR_i is the down ramp rate limit for the reactor i and UR_i is its upper ramp rate limit. Parameters $p_{i,0}$ are provided for each unit i. They refer to the production of the reactors at the start of first hour.

Other constraints are added to have a spinning reserve. This back-up is present to withstand some load variation and possible outage of units or networks components[24]. For the dynamic economic load dispatch problem, the spinning reserve can either be a fraction of the load or the capacity of the largest unit or their combination [24],[18].

$$\min(p_i^{max} - p_{i,t}; UR_i) \geq SR_{i,t} \quad \forall i, t \in \{1, \ldots, n\}, \{1, \ldots, m\}$$
$$\sum_{i=1}^{n} SR_{i,t} \geq SSR_t^{req} \quad \forall t \in \{1, \ldots, m\}$$

where $SR_{i,t}$ is a variable that gives the spinning reserve for the reactor i at time t and SSR_t^{req} is the required reserve parameter for a given hour t.

Hence the dynamic problem has the form :

$$\underset{p, SR \in \mathrm{R}^n}{\text{minimize}} \quad f(p) = \sum_{t=1}^{m} \sum_{i=1}^{n} a_i p_{i,t}^2 + b_i p_{i,t} + c_i + d_i |sin(e_i(p_{i,t} - p_i^{min}))| \tag{2.2a}$$

$$\text{subject to} \quad \sum_{i=1}^{n} p_{i,t} = D_t \ \ \forall t \in \{1, \ldots, m\} \tag{2.2b}$$

$$p_i^{min} \leq p_{i,t} \leq p_i^{max} \ \ \forall i, t \in \{1, \ldots, n\}, \{1, \ldots, m\} \tag{2.2c}$$

$$-DR_i \leq p_{i,t} - p_{i,t-1} \leq UR_i \quad \forall i, t \in \{1, \ldots, n\}, \{2, \ldots, m\} \tag{2.2d}$$

$$\min(p_i^{max} - p_{i,t}; UR_i) \geq SR_{i,t} \quad \forall i, t \in \{1, \ldots, n\}, \{1, \ldots, m\} \tag{2.2e}$$

$$\sum_{i=1}^{n} SR_{i,t} \geq SSR_t^{req} \quad \forall t \in \{1, \ldots, m\} \ . \tag{2.2f}$$

2.1.1 Surrogate problem

Regarding the transformation of the non-convex term, the sinusoidal part of the objective function, two different options have been tried to eliminate it in the new problem. The first one adds m new points at most at each iteration per reactor while the second one introduces new piece-wise under-approximation variables at the beginning but, at each iteration, one knot is created per interpolation if necessary. The second method produces results in less time.

In both cases, the variables of the surrogate problem request a new dimension. Indeed, while in the static case, the objective function summed the linear approximation over the piecewise linear interpolation and over the different reactors, in the dynamic one, it is added over those two dimensions and over the time one. Hence, we now have the variables $\eta_{i,t,j}$ and $\chi_{i,t,j}$. $\eta_{i,t,j}$ is the binary variable of the selection of the jth piecewise linear spline and $\chi_{i,t,j}$ is the electric production of the surrogate problem of the reactor i at the time t.

Method 1

In the first approach, the list of kink points used for the surrogate problem,for one reactor, holds over the whole timespan. The objective function and the constraints linked to the mixed integer programming can be written as:

$$
\begin{aligned}
\underset{p,\chi,\eta,SR}{\text{minimize}} \quad & g(p) = \sum_{i=1}^{n}\sum_{i=1}^{n}\left[a_i p_{i,t}^2 + b_i p_{i,t} + c_i + \sum_{j=1}^{n_i^{knot}-1}(\alpha_{i,j}\chi_{i,t,j} + \beta_{i,j}\eta_{i,t,j})\right] \\
\text{subject to} \quad & p_{i,t} = \sum_{j=1}^{n_i^{knot}-1} \chi_{i,t,j} \quad \forall i,t \in \{1,\dots,n\},\{1,\dots,m\} \\
& X_{i,j}\eta_{i,t,j} \le \chi_{i,t,j} \le X_{i,j+1}\eta_{i,t,j} \quad \forall i,j,t \in \{1,\dots,n\},\{1,\dots,n_i^{knot-1}\},\{1,\dots,m-1\} \\
& \sum_{j=1}^{n_i^{knot}-1} \eta_{i,t,j} = 1 \quad \forall i,t \in \{1,\dots,n\},\{1,\dots,m\} \\
& \eta_j \in \{0,\dots,1\}\ .
\end{aligned} \tag{2.3}
$$

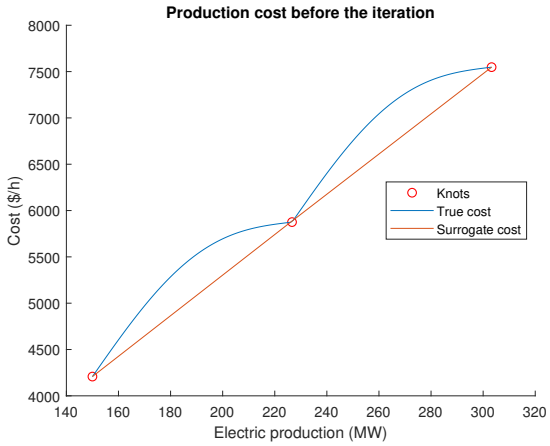
Hence, the set of nodes for a given reactor is still $X_{i,1} < \dots, X_{i,n_i^{knot}}$ and it is the same for every hour. The equation (2.3) is the same as in (1.7). For each time period, the piecewise linear spline interpolation are set with respect to the same set of knots. The update mechanisme corresponds to the following algorithm:

Algorithm 2 APQUA for dynamic case with one set of kink point for each time period

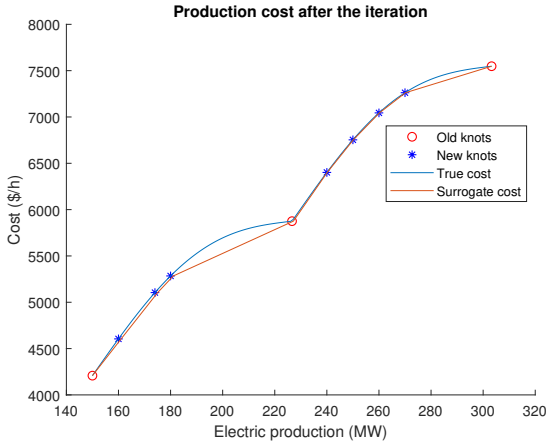
Set tolerance parameter $\delta_{\text{tol}} \ge 0$;
for $i = 1,\dots,n$ **do**
  Choose ordered knots $(X_{i,1},\dots,X_{i,n_i^{\text{knot}}})$ including the kink points $p_{i,1}^{\text{kink}},\dots,p_{i,n_i^{\text{kink}}}^{\text{kink}}$.
end for
repeat
  for $i = 1,\dots,n$ **do**
    for $j = 1,\dots,n_i^{\text{knot}} - 1$ **do**
      $\alpha_{i,j} \leftarrow (f^{\text{VPE}}(X_{i,j+1}) - f^{\text{VPE}}(X_{i,j}))/(X_{i,j+1} - X_{i,j})$;
      $\beta_{i,j} \leftarrow f^{\text{VPE}}(X_{i,j}) - \alpha_{i,j}X_{i,j}$;
    end for
  end for
  $\tilde{p} \leftarrow$ feasible (approximate) solution of the MIQP surrogate problem, obtained with an MIQP solver with absolute optimality gap tolerance γ;
  $\delta \leftarrow f(\tilde{p}) - g(\tilde{p})$;
  for $i = 1,\dots,n$ **do**
    for $t = 1,\dots,m$ **do**
      if $\min_{j\in\{1,\dots,n_i^{\text{knot}}\}} |\tilde{p}_{i,t} - X_{i,j}| > 0$ **then**
        $(X_{i,1},\dots,X_{i,n_i^{\text{knot}}+1})$
        $\leftarrow \text{sort}(X_{i,1},\dots,X_{i,n_i^{\text{knot}}},\tilde{p}_{i,t})$
        $n_i^{\text{knot}} \leftarrow n_i^{\text{knot}} + 1$;
      end if
    end for
  end for
until $\delta < \delta_{\text{tol}}$

The difference with respect to the static ELDP is the number of points added at each iteration. While previously one kink point χ was calculated, now there could be 24 new added points. If, for example, the optimum production for the generator i per hour is different than the points in the set of knots for 10 hours $j_1,\dots,j_{10}$ over the m ones, the set of knots before ordering is $\{X_{i,1},\dots,X_{i,n^{knot}},p_{i,j_1},\dots,p_{i,j_{10}}\}$. That drastically increases the number of variables over time.

Figure 2.1: Update mechanism



(a) Knots before one iteration



(b) Knots points after one iteration

Each point is added to the same set. An illustration of the update mechanism is given in the figures 2.1a and 2.1b.

Method 2

Another method consists of creating one set of points per reactor per time period. Hence, at the beginning, a large number of variables is created. Nevertheless, the number increases more slowly than in the previous approach. Before the start of the algorithm, each period for a certain reactor has the same set of knots. Those values are defined as for the static model.

$$n_{i,1}^{knot} = n_{i,2}^{knot} = \cdots = n_{i,m}^{knot} \ .$$

After one or more iterations, the number of knots is often different between the periods since the demand changes in function of time, which implies a specific behaviour per hour.

The kink points are defined by the same computation as in the static case. However, here, they are copied for each time period :

$$p_{i,t,j}^{kink} = (j-1)\frac{\pi}{e_i} + p_i^{min} \qquad n_{i,t}^{kink} = \left\lceil (p_i^{max} - p_i^{min})\frac{e_i}{\pi} \right\rceil + 1$$

$$p_{i,1,j} = p_{i,2,j} = \cdots = p_{i,m,j} \ .$$

The spline interpolation of the surrogate problem for all i, t with knots $X_{i,t,1} < X_{i,t,2} < ... < X_{i,t,n_{i,t}^{knot}}$ given by the above kink points is:

$$s_{i,t}(p_{i,t}) = \alpha_{i,t,j} p_{i,t} + \beta_{i,t,j} \ , p_{i,t} \in [X_{i,t,j}, X_{i,t,j+1}], \ j = 1, 2, \ldots, n_{i,t}^{knot} - 1 \tag{2.4}$$

where

$$\alpha_{i,t,j} = \frac{f_i^{VPE}(X_{i,t,j+1}) - f_i^{VPE}(X_{i,t,j})}{X_{i,t,j+1} - X_{i,t,j}}$$

is still the slope of the jth linear part and

$$\begin{aligned} \beta_{i,t,j} &= f_i^{VPE}(X_{i,t,j}) - \alpha_{i,t,j} X_{i,t,j} \\ &= \frac{X_{i,t,j+1} f_i^{VPE}(X_{i,t,j+1}) - X_{i,t,j} f_i^{VPE}(X_{i,t,j})}{X_{i,t,j+1} - X_{i,t,j}} \end{aligned}$$

Following the results presented in the first chapter, the mixed integer quadratic program surrogate problem can be written :

$$\underset{p,\chi,\eta,SR}{\text{minimize}} \quad g(p) = \sum_{i=1}^{n} \sum_{t=1}^{m} \left[a_i p_{i,t}^2 + b_i p_{i,t} + c_i + \sum_{j=1}^{n_{i,t}^{knot}-1} (\alpha_{i,j} \chi_{i,t,j} + \beta_{i,j} \eta_{i,t,j}) \right] \tag{2.5a}$$

$$\text{subject to} \quad \sum_{i=1}^{n} p_{i,t} = D_t \ \ \forall t \in \{1, \ldots, m\} \tag{2.5b}$$

$$p_{i,t} = \sum_{j=1}^{n_{i,t}^{knot}-1} \chi_{i,t,j} \quad \forall i, t \in \{1, \ldots, n\}, \{1, \ldots, m\} \tag{2.5c}$$

$$X_{i,t,j} \eta_{i,t,j} \leq \chi_{i,t,j} \leq X_{i,t,j+1} \eta_{i,t,j} \quad \forall i, t, j \in \{1, \ldots, n\}, \{1, \ldots, m\}, \{1, \ldots, n_{i,j}^{knot-1}\} \tag{2.5d}$$

$$\sum_{j=1}^{n_{i,t}^{knot}-1} \eta_{i,t,j} = 1 \quad \forall i, t \in \{1, \ldots, n\}, \{1, \ldots, m\} \tag{2.5e}$$

$$\eta_{i,t,j} \in \{0, \ldots, 1\} \tag{2.5f}$$

$$p_i^{min} \leq p_{i,t} \leq p_i^{max} \ \ \forall i, t \in \{1, \ldots, n\}, \{1, \ldots, m\} \tag{2.5g}$$

$$-DR_i \leq p_{i,t} - p_{i,t-1} \leq UR_i \quad \forall i, t \in \{1, \ldots, n\}, \{2, \ldots, m\} \tag{2.5h}$$

$$\min(p_i^{max} - p_{i,t}; UR_i) \geq SR_{i,t} \quad \forall i, t \in \{1, \ldots, n\}, \{1, \ldots, m\} \tag{2.5i}$$

$$\sum_{i=1}^{n} SR_{i,t} \geq SSR_t^{req} \quad \forall t \in \{1, \ldots, m\} \ . \tag{2.5j}$$

In this problem, the constraints (2.5d) and (2.5e) allow the selection of only one linear spline among the different available ones for a given reactor at a given time. Thanks to (2.5e), it is certain that the constraint (2.5e) will always be true except for only one linear spline between the nodes $[X_{i,t,j}, X_{i,t,j+1}]$.

The algorithm of the above surrogate problem is set as follows.

Algorithm 3 APQUA for dynamic case with several sets of kink points for each time period

1: Set tolerance parameter $\delta_{\text{tol}} \geq 0$;
2: **for** $i = 1, \ldots, n$ **do**
3: **for** $j = 1, \ldots, m$ **do**
4: Choose ordered knots $(X_{i,j,1}, \ldots, X_{i,j,n_i^{\text{knot}}})$ including the kink points $p_{i,1}^{\text{kink}}, \ldots, p_{i,n_i^{\text{kink}}}^{\text{kink}}$.
5: **end for**
6: **end for**
7: **repeat**
8: **for** $i = 1, \ldots, n$ **do**
9: **for** $j = 1, \ldots, m$ **do**
10: **for** $k = 1, \ldots, n_{ij}^{\text{knot}} - 1$ **do**
11: $\alpha_{i,j,k} \leftarrow (f^{\text{VPE}}(X_{i,j,k+1}) - f^{\text{VPE}}(X_{i,j,k}))/(X_{i,j,k+1} - X_{i,j,k})$;
12: $\beta_{i,j,k} \leftarrow f^{\text{VPE}}(X_{i,j,k}) - \alpha_{i,j,k} X_{i,j,k}$;
13: **end for**
14: **end for**
15: **end for**
16: $\tilde{p} \leftarrow$ feasible (approximate) solution of the MIQP surrogate problem, obtained with an MIQP solver with absolute optimality gap tolerance γ;
17: $\delta \leftarrow f(\tilde{p}) - g(\tilde{p})$;
18: **for** $i = 1, \ldots, n$ **do**
19: **for** $j = 1, \ldots, m$ **do**
20: **if** $\min_{k \in \{1, \ldots, n_{i,j}^{\text{knot}}\}} |\tilde{p}_{i,j} - X_{i,j,k}| > 0$ **then**
21: $(X_{i,j,1}, \ldots, X_{i,j,n_{i,j}^{\text{knot}}+1})$
 $\leftarrow \text{sort}(X_{i,j,1}, \ldots, X_{i,j,n_{i,j}^{\text{knot}}}, \tilde{p}_{i,j})$
22: $n_{i,j}^{\text{knot}} \leftarrow n_{i,j}^{\text{knot}} + 1$;
23: **end if**
24: **end for**
25: **end for**
26: **until** $\delta < \delta_{\text{tol}}$

This algorithm is different from the one in section 1.2.4. The time provokes now additional loops in the algorithm. Indeed, the nodes and the linear coefficients of the spline interpolation request now three dimensions instead of two as previously. In the line 17, g refers to the surrogate problem whereas f is the non convex one. g is now given by the set of nodes $\{X_{i,j,1}, \ldots, X_{i,j,n_{i,j}^{\text{knot}}}\}$.

The convergence of this algorithm towards the exact solution remains. The newly introduced constraints are all linear. They reduce the admissible domain but the objective function remains Lipschitz continuous on the dynamic economic load dispatch problem feasible set with a common Lipschitz constant K. The proof presented in [1] is still valid in this case.

2.2 Numerical results

2.2.1 Comparison between the methods on a basic case

Both presented algorithms have been tested on data extracted from [4]. It consists of ten different reactors with spinning reserve, ramp rate limit constraints. The production over different number of hours. Due to the high number of variables requested to perform the algorithms, those have been executed on the first five hours of the full dataset. Contrary to the other numerical simulations presented in this thesis, the computer used to test those methods is not ours. It is one of the clusters of the Université Catholique de Louvain. It is called Hmem, it has a 2.2 Ghz CPU and between 128 and 512 GB RAM.[7]. The figure 2.2 compares the run time with both

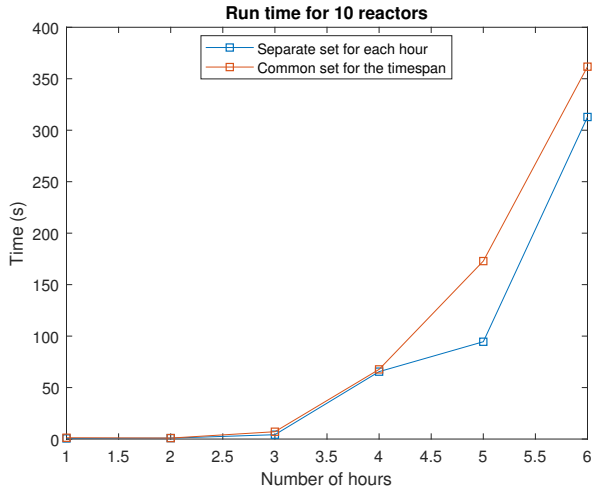


Figure 2.2: Comparison of run time between the model with a common set of nodes for the whole timespan and the model with a different set for each hour.

approaches.

In the second approach there is a preliminary step with the data. To have a data set more readable with a smaller size, only the nodes $X_{i,1,k}, \ldots, {}_{i,1,n^k_{i,1}}nots$ of the first time period and their number $m_{i,j}$ are stored in memory. This copying step takes 0.1s to get executed. This is only significative when the number of periods is smaller or equal to 2. For a number of hours larger than four, the model with sets of nodes corresponding to a given reactor and to a given hour runs faster. This procedure is used for the rest of the work.

Test over 24 hours

The second method has been applied to the dataset proposed in [4]. It consists of 10 reactors and 24 time periods. The spinning reserve requirement per hour is set to 5% of the load.

Due to the very high number of variables (more than 3,000, as shown in the figure 2.3), the algorithm 3 has been slightly modified at step 16 to limit the computation time to get a feasible solution of the MIQP surrogate problem. Otherwise, the number of iterations of the simplex algorithm used to solve this problem grows drastically and no feasible point is obtained within 10 minutes of computation. A limit has been imposed to the number of iterations for the simplex algorithm. This upper barrier of number of iterations leads the substitution problem to a non-optimal feasible solution $\hat{p}$. Hence, the approximation objective function of the surrogate problem $g(\hat{p})$ is higher or equals to the cost of the surrogate subproblem of the optimal point $g(\tilde{p})$. If the iteration limit is not high enough, the algorithm might converge towards a different solution from the optimal one. So, it is required to use a certain limit within a proportion with respect to the number of iterations if no barrier is set. Doing so, the approximate solution is very close to the optimal one.

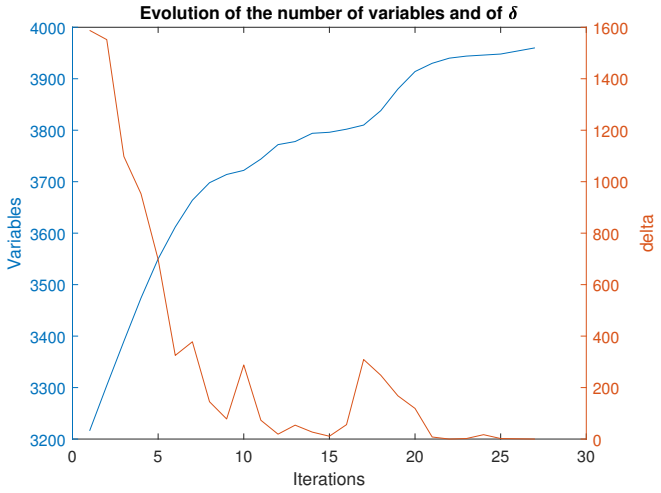


Figure 2.3: Evolution of δ and the number of variables for a 10-reactor and 24-hour set with a maximum simplex iteration of 2,000,000 .

The total production cost of the test case is 1,016,298 \$/day. The unit outputs are given in the table 2.1. Knowing the limitation of the simplex iterations, the computation time is 11,642.157 seconds. In the figure 2.3, details are shown about the numerical resolution on the previously mentioned data set. As this figure explains, the number of variables at the first step is 3216. Indeed, there are twice 240 variables which represent the production and the spinning reserve for the reactors and there are twice 1368 variables for the number of knots which is:

$$\sum_{i=1}^{n}\sum_{j=1}^{t} n_{i,j}^{knot} \ .$$

With this fact, it is obvious that the upper limit is necessary since the simplex algorithm worst-case complexity is exponential.

The cost can be compared to other results found in the literature. The lowest cost is mentioned in [24] and is also based on a MIQP algorithm, it is equal to 1,016,601 \$/day. That is larger than the one found in the present work. However, the procedure of [24] chooses to keep a good balance between the solution optimality and computation efficiency. Hence, there is no proof of convergence to the global optimum. This can explain the gap between those two techniques. There is also no proof of convergence for APQUA for this data since a value limits the iteration number of the MIQP solver.

The computation time makes the algorithm unpractical on bigger set. Indeed, test cases of hundreds of units are used in other approaches including, from time to time, loss term in the constraints. If that data was used with the current algorithm, the number of variables would be multiplied by 10. Hence, the MIQP solver would need more time to get a solution. A maximum limit of simplex iterations can also be imposed but due to the very high number of variables, the simplex algorithm requires more iterations than the limit in order to reach a point close to the optimum one. This would return a not so close point and the dispatch would differ widely to the optimal.

Table 2.1: Unit output power in MW for the test case unsing the dynamic APQUA algorithm

Hour	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10
1	150.000	143.560	185.533	60.000	122.867	122.450	129.590	47.000	20.000	55.000
2	150.000	223.560	229.399	60.000	73.000	122.450	129.590	47.000	20.000	55.000
3	150.000	303.560	297.399	60.000	73.000	122.450	129.590	47.000	20.000	55.000
4	226.624	316.799	305.670	60.000	122.867	122.450	129.590	47.000	20.000	55.000
5	226.624	396.799	299.670	60.000	122.867	122.450	129.590	47.000	20.000	55.000
6	303.248	396.799	303.399	70.415	172.733	122.450	129.590	54.364	20.000	55.000
7	303.248	396.799	297.399	120.415	172.733	122.450	129.590	84.364	20.000	55.000
8	379.873	396.799	297.399	166.710	122.867	122.450	129.590	85.312	20.000	55.000
9	456.497	396.799	294.373	191.246	172.733	122.450	129.590	85.312	20.000	55.000
10	456.497	396.799	303.399	241.246	222.600	160.000	129.590	85.312	21.556	55.000
11	456.497	396.799	297.399	291.246	222.600	160.000	129.590	85.312	51.556	55.000
12	456.497	460.000	307.614	291.329	222.600	160.000	129.590	85.312	52.057	55.000
13	456.497	396.799	302.815	241.329	222.600	160.000	129.590	85.312	22.057	55.000
14	456.497	396.799	297.399	191.329	172.733	122.450	129.590	82.201	20.000	55.000
15	379.873	396.799	297.399	162.270	122.867	160.000	129.590	52.201	20.000	55.000
16	303.248	396.799	294.642	112.270	73.000	122.450	129.590	47.000	20.000	55.000
17	226.624	396.799	297.399	62.270	122.867	122.450	129.590	47.000	20.000	55.000
18	303.248	396.799	299.866	70.415	172.733	125.036	129.590	55.312	20.000	55.000
19	379.873	396.799	293.827	120.415	172.733	122.450	129.590	85.312	20.000	55.000
20	456.497	460.000	312.586	170.415	222.600	160.000	129.590	85.312	20.000	55.000
21	456.497	389.533	322.603	120.415	222.600	122.450	129.590	85.312	20.000	55.000
22	379.873	309.533	283.094	70.415	172.733	122.450	129.590	85.312	20.000	55.000
23	303.248	229.533	204.000	60.000	122.867	122.450	129.590	85.312	20.000	55.000
24	226.624	222.266	189.757	60.000	73.000	122.450	129.590	85.312	20.000	55.000
Total cost ($/h)						1,016,298.1715				

Chapter 3

Multiple fuels

The economic load dispatch problem with multiple fuels and with valve-point effect is considered in this chapter. The introduction of multiple fuels does not change the convergence of the system. However, new constraints and new variables are introduced to support the new challenges brought by the several fuels available per generator. The first part of the chapter is about the model and how it works. The second part shows the numerical results obtained through simulations.

3.1 Model

3.1.1 Modified objective function and effects on the problem

Generators can employ different types of fuel to produce electricity. Hence the most economic fuel to burn has to be determined. In order to model this system, a "hybrid cost function" is used. Each dispatch unit is supplied with different fuel sources. Hence, they should be modelled thanks to several piecewise quadratic functions reflecting both the changes due to the fuel types and the selection of the fuel burned by the generator. Then, the fuel cost function is described as [8]:

$$f_i(p_i) = \begin{cases} a_{i,1}p_i^2 + b_{i,1}p^i + c_{i,1} + d_{i,1}|sin(e_{i,1}(p_{i,1} - p_i^{min}))|, \text{for fuel 1: } p_i^{min} \leq p_i \leq p_{i,1} \\ a_{i,2}p_i^2 + b_{i,2}p^i + c_{i,2} + d_{i,2}|sin(e_{i,k}(p_{i,2} - p_{i,1}))|, \text{for fuel 2: } p_{i,1} \leq p_i \leq p_{i,2} \\ \vdots \\ a_{i,k}p_i^2 + b_{i,k}p^i + c_{i,k} + d_{i,k}|sin(e_{i,k}(p_{i,k} - p_{i,k-1}))|, \text{for fuel k: } p_{i,k-1} \leq p_i \leq p_i^{max} \end{cases} \tag{3.1}$$

where $a_{i,k}, b_{i,k}$ and $c_{i,k}$ are the cost coefficients of the ith generator using the fuel of type k; $d_{i,k}$ and $e_{i,k}$ are other positive coefficients.

The cost function f_i remains continuous despite the possible change of fuel. Hence the maximum of power provided by the fuel i equals the minimum of power of the fuel j. This assures continuity.

The equation (3.1) is transformed into a different form which introduces new variables. For a certain reactor i, it gives:

$$f_i(p_i) = \sum_{j=1}^{n_i^{fuel}} a_{i,j}p_{i,j}^2 + b_{i,j}p_{i,j} + c_{i,j}u_{i,j} + u_{i,j}d_{i,j}|sin(e_{i,}(p_{i,j} - p_{i,j}^{min}))| \tag{3.2}$$

with n_i^{fuel} the number of fuels for the reactor i, $u_{i,j}$ a binary variable representing the use of the fuel j and $p_{i,j}^{min}$ the minimum power for the fuel j if it is used.

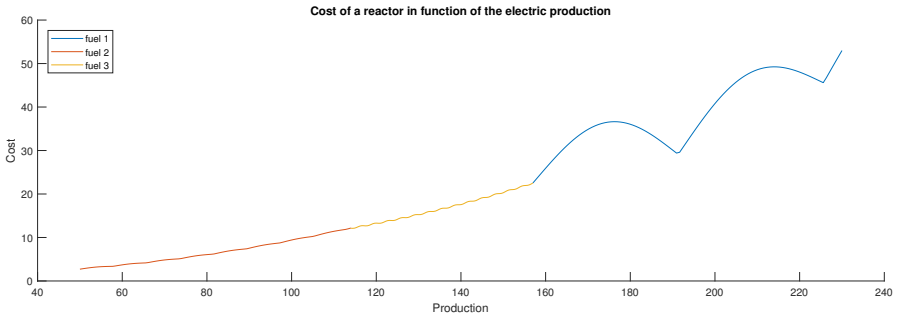


Figure 3.1: Cost of electric production for a reactor with three different fuels.

The equation above is the new objective function of the problem. However, the new set of constraints is also added; so, each reactor will select only one type of fuel. The approach is similar to the one used to select the piecewise function in the substitution problem 1.7.

$$p_i = \sum_{j=1}^{n_i^{fuel}} p_{i,j} \tag{3.3a}$$

$$p_{i,j}^{min} u_{i,j} \leq p_{i,j} \leq p_{i,j}^{max} u_{i,j} \quad \forall i, j \in \{1, \ldots, n\}, \{1, \ldots, n_i^{fuel}\} \tag{3.3b}$$

$$\sum_{j=1}^{n_i^{fuel}} u_{i,j} = 1 \quad \forall i \in \{1, \ldots, n\} \tag{3.3c}$$

$$u_{i,j} \in \{0, 1\} \tag{3.3d}$$

The constraint (3.5c) might seem useless but it is used in the MIQP reformulation. Furthermore, it gives a more general point of view to the problem. That set limits the use of fuel to only one by reactor. The others are set to zero.

Regarding the mixed-integer substitution problem, a similar problem, as found in the section **??** for the dynamic case, occurs. Indeed, each cost function f_i is composed of several parts. Every one of them has its own proper coefficients. The cost function for a reactor can be decomposed as:

$$f_i(p_i) = \sum_{i=1}^{n_i^{fuel}} f_{i,j}(p_{i,j})$$

So, when the problem is reformulated as a mixed-integer one, each function $f_{i,j}$ has to be reformulated into mixed-integer form $g_{i,j}$. It will be used by the APQUA algortihm. The non-convex term, the sine model of the valve-point effect, correspond to one spline reformulation. The function f_i is still Lipschitz continuous as shown in the figure 3.1 and only one knot, at most, will be added by reactor to the set of knots at each step of the algorithm. It is also supposed that the types of fuel are arranged in ascending order: the minimal production of fuel i is the maximum one of fuel j. This supposition has no incidence on the resolution of the surrogate problem by a solver but it has an impact on the proposed algorithm of the following section.

To make the model more realistic, limit ramp rate constraint, spinning reserves, as in the dynamic section, and prohibited zones can also be introduced. Those are place in the end of the model 3.5. About the prohibited operating zones, they represent units whose fuel characteristics are discontinuous. The feasible operating zones for a reactor i are defined by the set of equations [2]:

$$\begin{cases} p_i^{min} \leq p_i \leq p_{i,1}^{L} \\ p_{i,j-1}^{U} \leq p_i \leq p_{i,j} \\ p_{i,nz}^{U} \leq p_i \leq p_i^{max} \end{cases} \quad k = 2, 3, \ldots, nz_i \tag{3.4}$$

where nz_i is the number of prohibited zones for the unit i. A zone $z_{i,j}$ is the jth zone for the ith react and it is an interval defined by two limits such as : $z_{i,j} = [p_{i,j}^L, p_{i,j}^U]$. It is observed that each reactor having nz_i prohibited zones has $nz_i + 1$ disjoint operational zones. These disjoints parts form a non-convex set. Since the objective function remains Lipschitz-continuous, the APQUA algorithm is still proven to converge to the global optimum. The set (3.4) of equations is transformed under another form to be interpreted by an MIQP solver :

$$\begin{aligned} p_{i,j}^{OL}\zeta_{i,j} &\leq p_i\zeta_{i,j} \quad \forall i,j \in \{1,\dots,n\},\{1,\dots,nz_i+1\} \\ p_i\zeta_{i,j} &\leq p_{i,j}^{OU}\zeta_{i,j} \quad \forall i,j \in \{1,\dots,n\},\{1,\dots,nz_i+1\} \\ \sum_{j=1}^{nz_i} \zeta_{i,j} &= 1 \quad \forall i \in \{1,\dots,n\} \end{aligned}$$

with $\zeta_{i,j}$ is a binary variable saying if the production p_i belongs the feasible operating zone $[p_{i,j}^{OL}, p_{i,j}^{OU}]$. The figure 3.2 shows the feasible operating zones for a certain reactor. The objective function is still Lipstchiz-continuous while the domain is not anymore.

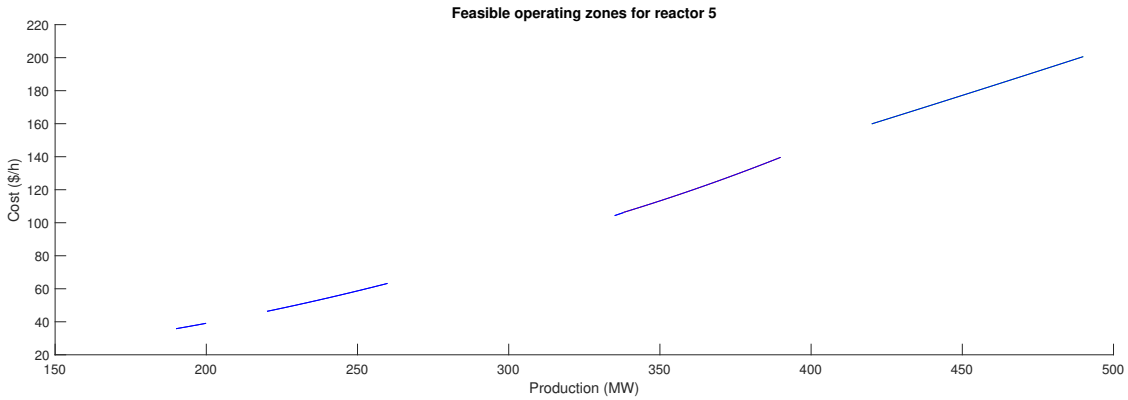


Figure 3.2: Feasible operating zones for the 5th reactor of the dataset of [2].

With those elements in mind, the MIQP substitution problem of the economic load dispatch problem with valve-point effect and multiple fuels is then written as :

$$\underset{p,\chi,\eta}{\text{minimize}} \quad g(p) = \sum_{i=1}^{n} \sum_{j=1}^{n_i^{fuel}} \left[a_{i,j} p_{i,j}^2 + b_{i,j} p_{i,j} + c_{i,j} u_{i,j} + \sum_{k=1}^{n_{i,j}^{knot}-1} (\alpha_{i,j,k} \chi_{i,j,k} + \beta_{i,j,k} \eta_{i,j,k}) \right] \tag{3.5a}$$

$$\text{subject to} \quad \sum_{i=1}^{n} p_i = D \tag{3.5b}$$

$$p_i = \sum_{j=1}^{n_i^{fuel}} p_{i,j} \tag{3.5c}$$

$$p_{i,j}^{min} u_{i,j} \leq p_{i,j} \leq p_{i,j}^{max} u_{i,j} \quad \forall i, j \in \{1, \ldots, n\}, \{1, \ldots, n_i^{fuel}\} \tag{3.5d}$$

$$\sum_{j=1}^{n_i^{fuel}} u_{i,j} = 1 \quad \forall i \in \{1, \ldots, n\} \tag{3.5e}$$

$$p_i = \sum_{j=1}^{n_i^{fuel}} \sum_{k=1}^{n_{i,j}^{knot}-1} \chi_{i,j,k} \quad \forall i \in \{1, \ldots, n\} \tag{3.5f}$$

$$X_{i,j,k} \eta_{i,j,k} \leq \chi_{i,j,k} \leq X_{i,j,k+1} \eta_{i,j,k} \quad \forall i, j, k \in \{1, \ldots, n\}, \{1, \ldots, n_i^{fuel}\}, \{1, \ldots, n_{i,j}^{knot} - 1\} \tag{3.5g}$$

$$\sum_{j=1}^{n_i^{fuel}} \sum_{j=1}^{n_i^{knot}-1} \eta_{i,j,k} = 1 \quad \forall i \in \{1, \ldots, n\} \tag{3.5h}$$

$$u_{i,j} \in \{0, 1\} \quad \forall i, j \in \{1, \ldots, n\}, \{1, \ldots, n_i^{fuel}\} \tag{3.5i}$$

$$\eta_{i,j,k} \in \{0, \ldots, 1\} \forall i, j, k \in \{1, \ldots, n\}, \{1, \ldots, n_i^{fuel}\}, \{1, \ldots, n_{i,j}^{knot}\} \ . \tag{3.5j}$$

$$p_i^0 - DR_i \leq p_i \leq p_i^0 + UR_i \quad \forall i \in \{1, \ldots, n\} \tag{3.5k}$$

$$\sum_{i=1}^{n} SR_i \geq SSR^{req} \tag{3.5l}$$

$$S_i = \begin{cases} \min\{(p_i^{max} - p_i), SR_i^{max}\}, & \forall i \in \{1, \ldots, n\} : nz_i = 0 \\ 0, & \forall i \in \forall i \in \{1, \ldots, n\} : nz_i > 0 \end{cases} \tag{3.5m}$$

$$p_{i,j}^L \zeta_{i,j} \leq p_i \zeta_{i,j} \quad \forall i, j \in \{1, \ldots, n\}, \{1, \ldots, nz_i + 1\} \tag{3.5n}$$

$$p_i \zeta_{i,j} \leq p_{i,j}^U \zeta_{i,j} \quad \forall i, j \in \{1, \ldots, n\}, \{1, \ldots, nz_i + 1\} \tag{3.5o}$$

$$\sum_{j=1}^{nz_i} \zeta_{i,j} = 1 \quad \forall i \in \{1, \ldots, n\} \tag{3.5p}$$

The constraints (3.5g), (3.5h), (3.5p) set every variable $\chi_{i,j,k}$, except one, of the set

$$[\chi_{i,1,1}, \ldots, \chi_{i,1,n_{i,1}^{knot}}, \chi_{i,2,1}, \ldots, \chi_{i,n_i^{fuel},n_{i,n_i^{fuel}}^{knot}}]$$

to zero. It leads to each $p_{i,j}$ being equal to zero except for only one. So, only one fuel is used by unit of production. This problem is convex since every constraint is linear. The objective function is assumed to be positive semidefinite. About the equation (3.5m), it is observed that units with prohibited operating zones limits the possibility of possible dispatch. Hence, the load regulation may lead to a prohibited zone. Consequently, an assumption is posed : units with prohibited operating zones do not participate to the spinning reserve of the system. [16]

3.1.2 Modified algorithm

The algorithm used to solve the ELDP with valve-point effect and with several types of fuel is:

Algorithm 4 APQUA for static case

1: Set tolerance parameter $\delta_{\text{tol}} \geq 0$;
2: **for** $i = 1, \ldots, n$ **do**
3: **for** $j = 1, \ldots, n_i^{fuel}$ **do**
4: Choose ordered knots $(X_{i,j,1}, \ldots, X_{i,j,n_i^{\text{knot}}})$ including the kink points $p_{i,j,1}^{\text{kink}}, \ldots, p_{i,j,n_i^{\text{kink}}}^{\text{kink}}$.
5: **end for**
6: **end for**
7: **repeat**
8: **for** $i = 1, \ldots, n$ **do**
9: **for** $j = 1, \ldots, n_i^{fuel}$ **do**
10: **for** $k = 1, \ldots, n_{i,j}^{\text{knot}} - 1$ **do**
11: $\alpha_{i,j,k} \leftarrow (f^{\text{VPE}}(X_{i,j,k+1}) - f^{\text{VPE}}(X_{i,j,k}))/(X_{i,j,k+1} - X_{i,j,k})$;
12: $\beta_{i,j,k} \leftarrow f^{\text{VPE}}(X_{i,j,k}) - \alpha_{i,j,k} X_{i,j,k}$;
13: **end for**
14: **end for**
15: **end for**
16: $\tilde{p} \leftarrow$ feasible (approximate) solution of the MIQP surrogate problem, obtained with an MIQP solver with absolute optimality gap tolerance γ;
17: $\delta \leftarrow f(\tilde{p}) - g(\tilde{p})$;
18: **for** $i = 1, \ldots, n$ **do**
19: **for** $j = 1, \ldots, n_i^{fuel}$ **do**
20: **if** $\min_{k \in \{1, \ldots, n_{i,j}^{\text{knot}}\}} |\tilde{p}_i - X_{i,j,k}| > 0$ **and** $\tilde{p}_i \in [X_{i,j,1}, X_{i,j,n_{i,j}^{knot}}]$ **then**
21: $(X_{i,j,1}, \ldots, X_{i,j,n_{i,j}^{\text{knot}}+1})$
 $\leftarrow \text{sort}(X_{i,j,1}, \ldots, X_{i,j,n_{i,j}^{\text{knot}}}, \tilde{p}_i)$
22: $n_{i,j}^{\text{knot}} \leftarrow n_{i,j}^{\text{knot}} + 1$;
23: **end if**
24: **end for**
25: **end for**
26: **until** $\delta < \delta_{\text{tol}}$

As explained earlier, new for loops are introduced in the algorithm due to the new dimension of the knots, i.e., the type of fuel. The rest of the algorithm performs in the same way as in section 1.2.4. A new condition appears in the line 20. The optimal production of the surrogate problem $\tilde{p}_i$ of the reactor i is added to only one set of knots; the one whose fuel is burned. The second condition prohibits other behaviours. Except for the knots representing the minimum and the maximum production for a reactor, the limit knots of intervals are present in two different constraints: the one in which they are the upper limit and the one in which they are the lower limit. That does not provoke an error. If $\tilde{p}$ is equal to a limit knot, it is not added to the set since the difference between the two points is equal to zero and the first condition is invalid.

3.2 Numerical tests

The reference data set for this kind of model is given in [8]. This data consists of ten reactors with two or three fuels. The algorithm 4 has been applied on that data. Two cases have been considered one without ramp rate limit, spinning reserve and prohibited operating zone and one with.

3.2.1 Test case with valve-point effect and multiple fuels

The results are in the table 3.1. In this publication the coefficients for the sine model are sometimes high. For example, the cost function for the ninth reactor for the second type of fuel contains 285 kink points. The table 3.1 also compares the decomposition obtained with a heuristic algorithm [19] whose total cost is one of the lowest known and whose objective function does not cause any inaccuracy due to using invalidated cost function [11]. The method presented in [19] is called DPD It is the abbreviation of DE(differential evolution)-PSO(particle swarm optimization)-DE(differential evolution). This model does not include ramp rate limit, spinning reserves and prohibited zone of operation constraints. The demand of the system equals 2700 MW.

Table 3.1: Simulation results for the test-system from [8], produced by DPD and APQUA.

Unit power output (MW)	Fuel	DPD	APQUA
U1	2	219.13200	218.59400
U2	1	212.40200	211.71174
U3	1	280.65700	280.65706
U4	3	239.28300	239.63943
U5	1	279.93500	279.93452
U6	3	239.79500	239.50506
U7	1	287.72700	287.72749
U8	3	239.41800	239.77380
U9	3	425.78200	426.58829
U10	1	275.86900	275.86861
Total cost ($/h)		623.826930	623.82263
Computation time (s)		1.655	634.222

The APQUA algorithm generates a cost near the other one. It is even smaller, which is due to the global convergence. In the review paper [11], a cost near 623.83$/h is expected, costs way lower than that are due to inaccuracy in the objective function. The output of the APQUA method is slightly below the limits but the margin is small. This behaves as expected and the returned results are better than other presented in the review publication. However, the computation time is more than 300 times superior for the APQUA approach with respect to the DPD heuristic technique.

3.2.2 Test case with valve-point effect,multiple fuels, ramp-rate limits, spinning reserve and prohibited operating zones

The algorithm 4 as also been tested on a set whose reactor data comes from [8] and sinning reserve and prohibited zones data are originated from [2]. The output is presented in the table 3.2. The production and cost are compared with the ones of [19]. The demand is 2700 MW and the minimum spinning reserve for the system is 120 MW.

The returned cost by the APQUA algorithm is also lower than the one proposed by the DPD. Regarding the compution time, it is more than 300 times greater for the APQUA method.

Those two cases confirm the global convergence of APQUA and proposed new lower bounds for the datasets.

Table 3.2: Simulation results for the test-system from [8] and [2], produced by DPD and APQUA.

		DPD			APQUA	
Unit	Fuel	S_i (MW)	Production(MW)	Fuel	S_i (MW)	Production (MW)
U1	2	28.81100	221.10500	2	31.75117	218.24883
U1	2	28.81100	221.10500	2	29.35200	220.64800
U2	1	17.10300	212.65960	1	17.05044	212.94956
U3	1	0.00000	283.65710	1	0.00000	283.68170
U4	3	24.91100	240.68640	3	24.55435	240.44565
U5	1	0.00000	260.00000	1	0.00000	260.00000
U6	3	24.66600	240.66100	3	24.55435	240.44565
U7	1	0.00000	292.72750	1	0.00000	292.46952
U8	3	24.37000	240.55200	3	24.55435	240.44565
U9	3	1.10600	438.08280	3	1.08574	438.91426
U10	1	0.00000	269.86860	1	0.00000	270.00000
Total cost ($/h)			624.4274			624.33149
Computation time (s)			2.09438			702.4729

Chapter 4

Power loss term

In this chapter, the effect of the introduction of a loss term in the load constraint is investigated. The changes on the model and its behaviour are explained and some guidelines to solve the new issues are shown. Some numerical results and comparisons with reference datasets are also given to see if the outputs match the expectations.

4.1 Model

During the creation and the transport of the electricity some parts are lost. It is due to several reasons. The network can lead to losses because its wires are not isnsulated enough. During the production, the wires are heated and some of the electricity is lost in heat because of the Joule effect. Those events are linked to the electric consumption. The more energy is produced, the more the losses growth. Consequently, this changes the load constraint of our problem. It is now written as :

$$\sum_{i=1}^{n} p_i - p_L(p) = D \ . \tag{4.1}$$

Whereas previously, the term $p_L(p)$ was considered to be equal to 0, it is not the case anymore. It has the following from, as proposed in [12]:

$$p_L(p) = \sum_{i=1}^{n}\sum_{j=1}^{n} p_i B_{i,j} p_j + \sum_{i=1}^{n} p_i B_{0i} + B_{00} \tag{4.2}$$

where $B_{i,j}$, B_{0i} and B_{00} are loss coefficients. The terms $B_{i,j}$ belong to a symetric matrix.

The loss coefficients values can be given in megawatt, per unit value, but also in MVAbase. The transforamtion to switch between unit is given in appendix A.3.

The introduction of a non-zero transmission loss term makes the constraint (4.1) non linear due to the matrix product around the coefficient matrix B. An MIQP solver can no longer be used to solve this problem. Some nonlinear solvers, as knitro, are able to solve those kind of optimisation problems but the global convergence is no longer assured; only a local optimality is guaranteed. Hence, the perk of APQUA is lost. To continue to use a MIQP solver, one can linearize the quadratic equality constraint. An approximate global solution is found to the new problem and the error is limited. There exist different methods to handle this issue. Some are presented and tested in the next sections.

The equation (4.1) is an quadratic equality constraint. It makes the problem non convex even when the valve-point effect is neglected in the objective function (1.3a). That change modifies the domain of feasibility. Previously, it was a polyhedron. It transforms to an ellipsoidal under the ELDP with power-loss term. However, the ellipsoid may be way bigger then the limits of the domain as shown in the figure 4.1.

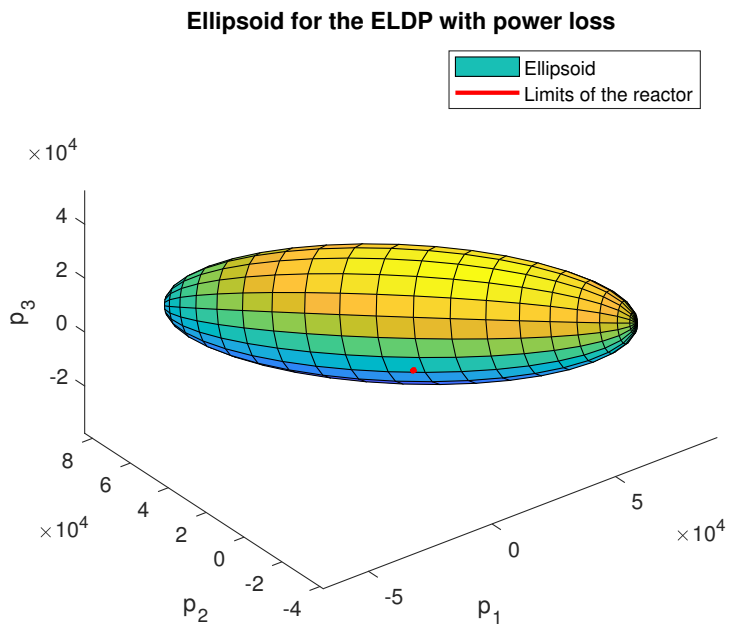


Figure 4.1: Ellipsoid of the feasible domain for the ELDP with power loss term for three reactors. The maximum and minimum power of the generator are located on the red dot.

A visualization of the domain for two variables is given at the figure 4.2. From the two figures presented in this section several observations can be extracted. The ellipsoid being way larger than the polyhedron defined by the upper and lower power limits of the reactor, an precise approximation of that shape by an set of plane restricted to the feasible domain would not yield to a lot of differences for the objective function.

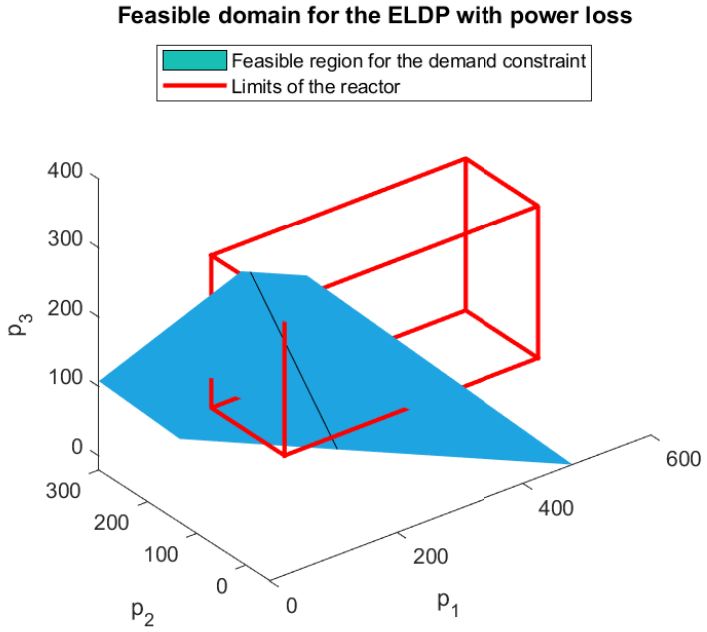


Figure 4.2: Feasible domain for the ELDP with power loss term for three reactors. The maximum and minimum power of the generator are on the polytope. The point that satisfy the demand equality constraints are located on the blue curbe.

4.1.1 Reformulation linearization technique

The idea behind this approach is to introduce new variables $Y_{i,j} = p_i p_j$ to replace the product $p_i p_j$[3],which relaxes the quality equation. The condition (4.1) becomes linear and a classic MIQP solver can be used to solve the surrogate problem. In this section, the principle of the reformulation linearization technique (RLT) is applied to the optimal power dispatch problem with power loss term.

To simplify the expressions, the scalar product between two matrices is introduced

$$A \cdot B = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{i,j} B_{i,j} \ .$$

The transformation of the surrogate problem to a general form for a quadratically constrained quadratic programming problem (QCQP) is:

$$\begin{aligned}
&\max_{p \in \mathrm{R}^n} && p^T Q_0 p + a_0^T p + k = -f(p) && \text{(4.3a)}\\
&\text{subject to} && pQp^T + a^T p = \hat{p_D} && \text{(4.3b)}\\
& && p_i^{min} \leq p_i \leq p_i^{max} \ \ \forall i \in \{1, ..., n\} \ . && \text{(4.3c)}
\end{aligned}$$

where p is the vector of the electric production of the units, $Q = B$, $a^T = -B_0(B' + B) + 1^T$, $\hat{p_D} = D + B_{00}$ from (4.2).

The RLT relaxation of a quadratically constrained qudratic problem QCQP is based on products of upper and lower bound constraints of the original variables to obtain valid linear inequality constraints on the new variables Y_{ij}.

For two variables p_i and p_j the limitations are

$$\begin{aligned}
p_i - p_i^{min} &\geq 0 & \qquad p_i^{max} - p_i &\geq 0\\
p_j - p_j^{min} &\geq 0 & \qquad p_j^{max} - p_j &\leq 0 \ .
\end{aligned}$$

By Multiplying all of the constraints located above associated to p_i with a constraint involving p_j , and by replacing the product term $p_i p_j$ from (4.3b) with the new variable $Y_{i,j}$, the constraints are obtained

$$Y_{i,j} - p_i^{min} p_j - p_j^{min} p_i \geq -p_i^{min} p_j^{min} \quad (4.5)$$

$$Y_{i,j} - p_i^{max} p_j - p_j^{max} p_i \geq -p_i^{max} p_j^{max} \quad (4.6)$$

$$Y_{i,j} - p_i^{min} p_j - p_j^{max} p_i \leq -p_i^{min} p_j^{max} \quad (4.7)$$

$$Y_{i,j} - p_j^{min} p_i - p_i^{max} p_j \leq -p_i^{max} p_i^{min} \quad (4.8)$$

with $i, j = 1, ..., n$. Those constraints also hold when $i = j$.In that situation, the last two constraints are identical. Furthermore, when the condition $Y_{i,j} = Y_{j,i}$ is set, the last two constraint are equal for each i, j. Hence, the relaxation of the surrogate problem of ELDP with power-loss term can be written

$$\begin{aligned}
\max \quad & -\hat{g}(p) = \sum_{i=1}^{n} \left[a_i Y_{i,i}^2 + b_i p_i + c_i + \sum_{j=1}^{n_i^{knot}-1} (\alpha_{ij}\chi_{ij} + \beta_{ij}\eta_{ij}) \right] && \text{(4.9a)} \\
\text{subject to} \quad & Q \cdot Y + a^T p = \hat{p_D} && \text{(4.9b)} \\
& p_i^{min} \leq p_i \leq p_i^{max} \ \ \forall i \in \{1, ..., n\} && \text{(4.9c)} \\
& p_i = \sum_{j=1}^{n_i^{knot}-1} \chi_{ij} && \text{(4.9d)} \\
& X_{i,j}\eta_{i,j} \leq \chi_{i,j} \leq X_{i,j+1}\eta_{i,j} && \text{(4.9e)} \\
& \sum_{j=1}^{n_i^{knot}-1} \eta_{ij} = 1 && \text{(4.9f)} \\
& \eta_j \in \{0, ..., 1\} && \text{(4.9g)} \\
& Y - p^{min}p^T - p(p^{min})^T \geq -p^{min}(p^{min})^T && \text{(4.9h)} \\
& Y - p^{max}p^T - p(p^{max})^T \geq -p^{max}(p^{max})^T && \text{(4.9i)} \\
& Y - p^{min}p^T - p(p^{max})^T \leq -p^{min}(p^{max})^T && \text{(4.9j)} \\
& Y = Y^T && \text{(4.9k)}
\end{aligned}$$

Since $Y_{i,j} = Y_{j,i}$, the above problem is a linear one[3].

In the later part, the constraint (4.1) is relaxed. Hence, the problem is now a quadratically constrained quadratic programming (QCQP) problem without any quadratic equality constraint. So, an MIQP can be used to solve it.

4.1.2 Approximation by tangent planes

This approach, presented in [5], consists in modifying the quadratic equality constraint induced by the power loss term. The feasible set of points verifying the equality is changed to a polyhedron. This linearization aims to replace the quadratic constraint by M different linear constraints. There exists a convex polytope that covers any given convex set in the Euclidean plane. Let $\mathcal{P}$ be problem given in the equation (4.1). $\mathcal{S}$, the ellipsoid, is described, in a general form, by:

$$\mathcal{S} := \{x \in \mathrm{R}^n : (x-b)^T B(x-b) \leq \tilde{b}\} \ .$$

Let ∂S be the boundary of the ellipsoid S. Then, a set of M points, $\chi_M = \{x_1, \ldots, x_M\}$ such that every $x_j \in \partial S$ for $j \in \{1, \ldots, M\}$, is generated to get M linear constraints to replace this particular quadratic constraint. Those points are obtained thanks to a lower discrepancy method detailed later. With those M different points, the set of constraints is

$$(x-b)^T B(x_j - b) \leq \tilde{b} \ \forall j \in \{1, \ldots, M\} \ .$$

It can be observed that these planes are tangent to S at the points of the set χ_M. In the figure 4.3, a set τ constituted of 5 linear constraints is showed. It is constructed around a ellipsoidal feasible set in two dimensions. Using, those M linear constraints, an approximate problem $\mathcal{P}(\chi_M)$ can be written as:

$$\begin{aligned}
\max_{p \in \mathrm{R}^n} \quad & p^T Q_0 p + a_0^T p + k \\
\text{subject to} \quad & (p-b)^T B(x_j - b) \geq \tilde{b} \ \ \forall j \in \{1, \ldots, M\} \\
& p_i^{min} \leq p_i \leq p_i^{max} \ \ \forall i \in \{1, ..., n\} \ .
\end{aligned}$$

Instead of solving the original problem, its approximation $\mathcal{P}(\chi_M)$ is solved for a large enough value of M. Hence, the accuracy of the approximation is correlated to the number of points in χ_M.

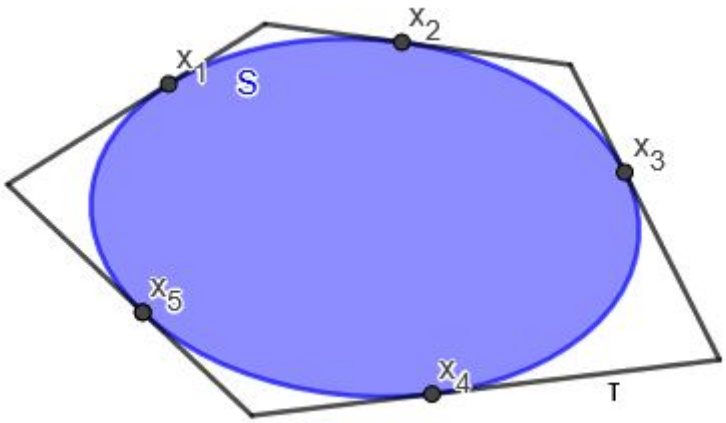


Figure 4.3: The tangent plane τ is an approximation of S with the 5 tangent points $x_1, \ldots, x_5$.

The precision of the solution of the approximate problem exclusively relate on the choice of $\mathcal{P}(\chi_M)$. The tangent planes to S at the M points create a cover of S. The type of cover is the bonded cover. It is defined as:

Let τ be the convex polytope generated by the tangent planes to S at the points $x_1, \ldots, x_M \in \partial S$. The polytope is said to be a bounded cover of S if,

$$d(\tau, S) = \sup_{t \in \tau} d(t, S) < \infty$$

where $d(t, S) = \inf_{x \in S} \|t - x\|$ and $\|\cdot\|$ represents the Euclidean distance.

There exists a bounded cover with only $M + 1$ points[5]. The goal is to find a bounder cover extremely close to the ellipsoid since it leads to a better approximation. Sadly, an arbitrary selection of points can surround $\mathcal{S}$ such that the feasible set can expand widely. It might also create an unbounded cover. So, the set χ_M has to generate to an optimal bounded cover. Mathematically a set $\tau^* = (x_1^*, \ldots, x_M^*)$ is an optimal bounded cover if

$$\sup_{t \in \tau^*} d(t, S) \leq \sup_{t \in \tau} d(t, S)$$

for any bounded cover τ created by any other sets of points of size M. The set $\{x_1^*, \ldots, x_M^*\}$ is declared to be the optimal M-point set. This set can also be written as the set of M points minimizing the maximum distance between τ and S:

$$\tau^* = \underset{\tau}{argmin}\ d(\tau, S)\ .$$

.

The optimal M-point set τ^* on the unit circle in two dimensions are the M-th roots of unity, unique up to rotation. This set also has the property of minimizing the discrete Riesz energy for the unit circle. The Riesz energy of a point set $R_M = \{x_1, \ldots, x_M\}$ is defined as $E_s(R_M) = \sum_{i \neq j = 1}^{M} \|x_i - x_j\|^{-s}$ for a positive real parameter s. A set which minimizes the Riesz energy on ∂S converges to the normalized surface measure of ∂S[14].

So, the optimal M-point τ^* is a set correlated to that set of M points which minimize the Riesz energy on ∂S. To obtain those kind of configuration of points, those have to be equidistributed. It is done in two step : first a good set on the unit hypercube is selected; then it is mapped to ∂S such that the equidistribution property still holds.

A set of points is equidistributed if the expected number of points inside any axis-parallel subregion matches the true number of points. This type of set in $[0,1]^n$ is called (t,m,n)-net in base μ. It is defined as a set of $M=\mu^m$ points in $[0,1]^n$ where any axis parallel μ-adic box with volume μ^{t-m} would contain exactly μ^t points. It is referred a low-discrepancy set [10]. There exist different methods to construct these point sets. The one used here is base on the Halton sequence[15].

Now, the set obtained thanks to the sequence is mapped firstly to and hyper-sphere $\mathrm{S}^n = \{x \in \mathrm{R}^{n+1} : x^T x = 1\}$ then secondly to ∂S.

The cylindrical coordinates of the n-sphere are wirtten as

$$\begin{aligned} x = x_n &= (\sqrt{1-t_n^2}x_{n-1}, t_n) \\ &\vdots \\ x_2 &= (\sqrt{1-t_2^2}x_1, t2) \\ x_1 &= (cost\phi, sin\phi) \end{aligned}$$

where $0 \leq \phi \leq 2\pi, -1 \leq t_k \leq 1, x_k \in \mathrm{S}^d$ and $k = (1,\ldots,n)$. Hence, a given point $x \in \mathrm{S}^n$ is characterized by an angle ϕ and heights $t_2,\ldots,t_n$ as:

$$x = x(\phi, t_2, \ldots, t_n), \qquad \text{with } 0 \leq \phi \leq 2\pi, -1 \leq t_1, \ldots, t_n \leq 1 \ .$$

A point $y = (y_1,\ldots,y_n) \in [0,1)^n$ is mapped to $x \in \mathrm{S}^n$ by

$$\begin{aligned} \varphi_1(y_1) &= 2\pi y_1 \\ \varphi_k(y_k) &= 1 - 2y_k \forall k \in \{2,\ldots,n\} \end{aligned}$$

and by cylindrical coordinates $\mathbf{x} = \mathbf{\Phi_n}(\mathbf{y}) = \mathbf{x}(\varphi_1(y_1), \varphi_2(y_2), \ldots, \varphi_n(y_n))$. The mapping $\Phi_n : [0,1)^n \to \mathrm{S}^n$ preserve the area[6].

Regarding the mapping from the point set from S^{n-1} to ∂, it is defined by ψ. It gives:

$$\psi(x) = \sqrt{\tilde{b}} B^{-1/2} x + b \ .$$

In [5], the mapping is shown to preserve the area. An example of mapping is given in the figures 4.4 and 4.5. The first one references to the mapping from a $(0, 40, 2)$-net set in base to the unit sphere in R^3. The points are the red points. The second figure show the mapping to the ellipsoid. The points are located all around the shape. So, the algorithm to obtain the set of points $x_1,\ldots,x_M$ used to approximate the optimization problem (4.1) in detailed in the algorithm 5. Once the points are obtained, the approximate problem $\mathcal{P}(\chi_M)$ is solve. It converges to $\mathcal{P}$ as $N \to \infty$[5].

In the present problem, the ellipsoid is encapsulated inside the set of plane as seen in the figure 4.2. Given the size of the ellipsoid with respect to the limits of the reactor, it is possible that no point of the (t,m,n) set is located within the feasible domain. The problem would then be unfeasible. This prohibits the usage of an equality instead of an inequality in the equation (4.10).

Furthermore, the present case want to minimize around the polyhedron. The feasible area is the exterior of the polytope. When points are place all around the surface, the problem becomes unfeasible since there exists no point verifying all the conditions of the equation (4.1.2). The points have to be place on only one part of the shape. The goal would be to only select points within the feasible domain given by the lower and upper production limits of the reactors. It would request to find the equation of the surface being the intersection of the ellipsoid and the feasible domain. From that, cylindrical coordinates could be extracted. During the mapping on the hypersphere, $\phi, t_1, \ldots, t_n$ would be restricted to smaller intervals than $[0, 2\pi]$ and $[-1, 1]$.

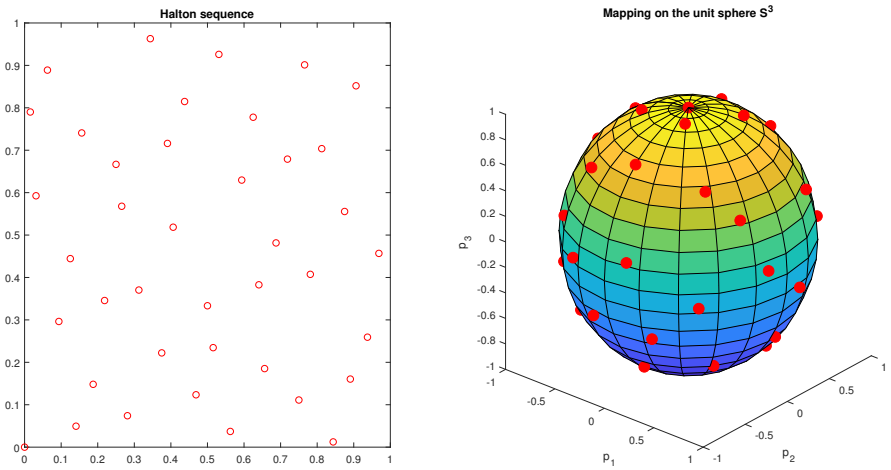


Figure 4.4: Mapping from a $(0, 40, 2)$-net to the unit sphere in R^3

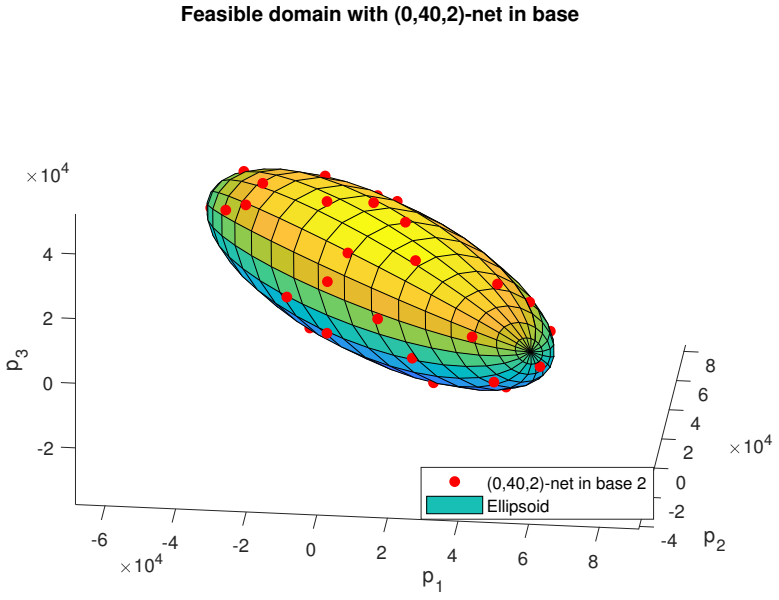


Figure 4.5: Caption

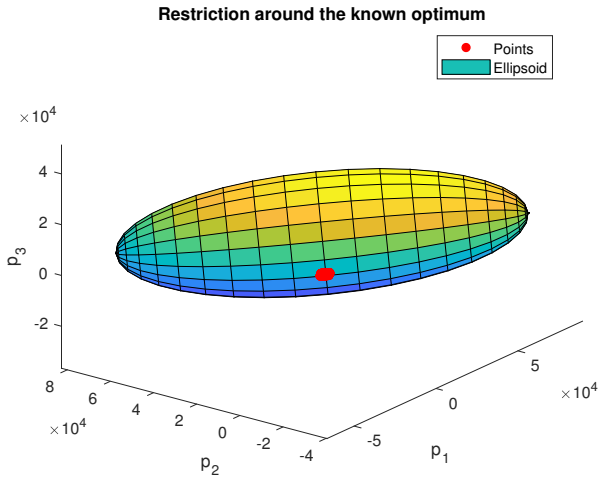


Figure 4.6: Positions of the points for the approximation of the ellipsoid by tangent planes.

Algorithm 5 Point simulation on ∂S

Input : $B, b, \tilde{b}$ to specify S and $M = \eta^k$ points
Output : $x_1, \ldots, x_M \in \partial S$
Generate $y_1, \ldots, y_M$ as a $(t, k, m-1)$-net in base η
for $i = 1, \ldots, M$ **do**
    $x_i = \phi \circ \Phi_{m-1}(y_i)$
end for
return $x_1, \ldots, x_N$

4.2 Numerical applications

The approaches presented in the precedent section have been applied on two data sets based on the one proposed in [13].

In the first case, the power-loss term, prohibited zones and ramp rate limit are taking into account but the valve-point effect is ignored in order to analyze the effect of loss term. The set has 6 different reactors and the demand is 1263 MW. To compare the approximations introduced by the method presented in this section, a branch and bound algorithm, which is proven to globally converge up to an suboptimal point[17], and a MIQP solver have been applied. For the branch and bound algorithm, the load constraint (4.1) is set to and equality whereas, for the MIQP, it is an inequality. Then, if coefficient matrix are positive semidefinite, the problem is a convex quadratically constrained quadratic program. It is the case, the technique convergence to the global optimum. A trick is used regarding the approximation by plane. An optimum is found by the other methods. Then, the low discrepancy sequence is plot on the unit sphere nearby that point. The figure 4.6 So, feasible points can be found.

In the second test case, the objective function considers the valve-point effect and the others constraints used in the first case are also applied. The demand stays 1263 MW and the same algorithms are applied. The parameters of the system are based on [13]. The coefficients d_i,e_i linked to the valve-point effect are arbitrary such that:

$$d_i = 0.005 \; e_i = 0.05 \; \forall i \in \{1, \ldots, n\} \; .$$

This test case for the table 4.1 uses a reference set from [11]. The estimated global optimal cost should be around 19449.9 \$/h, which is indeed the one obtain thanks to the first two methods. The relaxation linearization technique generates a cost which underapproximate the real one,

Table 4.1: Output of the different methods for the first test case, valve-point effect neglected.

	MIQP	Branch&Bound	RLT	Plane
U1	447.507	447.504	448.630	320.000
U2	173.319	173.318	174.208	80.000
U3	263.460	263.463	262.806	100.000
U4	139.068	139.065	140.744	60.000
U5	165.471	165.473	163.754	100.000
U6	87.132	87.135	85.000	50.000
Total cost ($/h)	15449.900	15449.900	15439.717	8468.750
Loss (MW)	12.958	12.958	12.902	4.668
Run time (s)	0.031	10.219	0.047	0.437

this is due to the constraints not being tight enough. The same effect appears for the tangent plane approximation.

Table 4.2: Output of the different methods for the second test case, valve-point effect is not neglected

	MIQP	Branch&Bound	RLT	Plane
U1	450.744	450.767	450.600	380.000
U2	175.718	175.701	175.662	80.000
U3	265.000	265.000	264.332	100.000
U4	141.558	141.541	144.248	60.000
U5	167.976	167.989	165.456	150.000
U6	75.000	75.000	75.000	50.000
Total cost ($/h)	15451.608	15451.608	15443.397	9807.771
Loss (MW)	12.997	12.997	12.912	6.718
Run time (s)	0.516	27.641	0.406	0.891

Same results as the other case are returned. The relaxation of the quadratic constraint has no impact on the convergence of the algorithm since the both the objective function and the matrix

$$-pBp^T - p(B_o + \mathbf{1})^T - Boo$$

from the equation (4.1) are semidefinite positive. This explains why the outputs of the MIQP solver and the branch & bound solver are equivalent regarding the objective cost. However, the MIQP approach converges 50 times faster than the other one. The other two techniques are subject to the same errors as previously.

Conclusion

This thesis presented a numerical analysis and an expansion of the new approach from [1].

First, the given method was applied on a reference dataset. As expected, the generation cost returned by the algorithm performed well: the proven optimization bound quoted in review papers was reached despite being a complex set with several local minimums. Also, the run time was inferior to five seconds, which makes the method usable in day to day work.

In the second part, the algorithm has been expanded to the dynamic economic load dispatch problem by two different approaches where the mixed integer reformulation is made differently. The first one proposed to have common node set for each hour. A maximum of 24 knots per reactor were then added. The second suggested to create one knot set by hour for each reactor. Only one knot, maximum, was introduced for each set. The second method performs better in run time for a higher number of initial variables,when the repartition is planned over more hours. However, due to the high number of variables, thousands of them, the run time over 24 hours was superior to several hours despite being executed on supercomputers of the university, which is much larger than others reported by other papers. Despite that, even when a maximum number of iterations was imposed on the MIQP solver, the returned cost was within a small range of heuristic methods. A longer run would have reached a zero gap between the mixed integer problem and the real one.

Then, new constraints have been added to the model. The electricity generation facilities had several disposable fuels to meet the demand. In addition, prohibited operation zones, spinning reserve and ramp-rate limits constraints are introduced. Those new constraints do not modify the objective function which remains convex for the mixed-integer relaxation. The studied technique is then applied, among others, to a reference dataset. The results are slightly better, i.e., the objective function is lower. The run time is limited by the introduction of new constraints and variables, it was the same problem for the dynamic version of the problem.

Finally, the power-loss term problem is tackled. The electricity losses during the generation are considered in the model. The APQUA method does no longer guarantee a convergence to the global minimum. Indeed, the load equality constraint is now quadratic. The substitution problem is not convex anymore and a mixed-integer solver is not usable. Different relaxation methods were proposed: relaxation linearization technique, tangent plane of the ellipsoid feasible domain obtained by low discrepancy sequences or relaxation of the equality constraint. They were then numerically evaluated and compared with a branch and bound algorithm, which converges to a global minimum at the cost of a long effort. Whereas the relaxation of the demand constraint generates output results near the branch and bound ones since both the objective function and the relaxed constraints are convex. The output of the two other approaches differ significantly; especially for the approximation by tangent planes whose output underestimate the objective function.

Through this thesis, the algorithm has been shown to reach optimal minimum, to lower the optimization bounds. As expected, the drawback is the run time. However, a fast speed of convergence is not the goal of the technique.

Further works could focus on different aspects. A possible selection of knots among the available ones should be investigated. A restriction would reduce the number of variables in the algorithm and then speed up the convergence, making it usable for larger data. It should be done

while keeping the global convergence property of the APQUA technique. The restriction might be based on the selection of only some units or on the exclusion of some parts, some regions, of the production capacity based on where the previous iteration finished.

Regarding, the power-loss term other researches could explore the tangent planes relaxation. In this thesis, an approximation of the whole ellipsoid by tangent planes has been proposed. It would be interesting to narrow it to the part which is contained within the production limits of each reactor since most of the points created by the method are useless. They are located outside the feasible production capacity. This would lead to a better approximation of the cost by reducing the gap, within the feasible domain, between the ellipsoid and its relaxation to a polyhedron.

Reference

[1] P.-A. Absil, B. Sluysmans, and Stevens N. “MIQP-Based Algorithm for the Global Solution of Economic Dispatch Problems with Valve-Point Effects”.

[2] Nima Amjady and H Nasiri-Rad. “Economic dispatch using an efficient real-coded genetic algorithm”. In: 3 (Apr. 2009), pp. 266–278.

[3] Kurt M. Anstreicher. “Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming”. In: *Journal of Global Optimization* 43.2 (Mar. 2009), pp. 471–484. ISSN: 1573-2916. DOI: 10.1007/s10898-008-9372-0. URL: https://doi.org/10.1007/s10898-008-9372-0.

[4] P. Attaviriyanupap et al. “A hybrid EP and SQP for dynamic economic dispatch with nonsmooth fuel cost function”. In: *IEEE Trans. Power Syst.* 17.2 (2002), pp. 411–416.

[5] K. Basu, A. Saha, and S. Chatterjee. “Large-Scale Quadratically Constrained Quadratic Program via Low-Discrepancy Sequences”. In: *ArXiv e-prints* (Oct. 2017). arXiv: 1710.01163 [stat.ML].

[6] Johann S. Brauchart and Josef Dick. “Quasi–Monte Carlo rules for numerical integration over the unit sphere $${\mathbb{S}^2}$$”. In: *Numerische Mathematik* 121.3 (July 2012), pp. 473–502.

[7] Céci. *Clusters at Céci*. 2017. URL: http://www.ceci-hpc.be/clusters.html (visited on 06/08/2018).

[8] Chao-Lung Chiang. “Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels”. In: *IEEE Transactions on Power Systems* 20.4 (Nov. 2005), pp. 1690–1699. ISSN: 0885-8950. DOI: 10.1109/TPWRS.2005.857924.

[9] Keely L. Croxton, Bernard Gendron, and Thomas L. Magnanti. “A Comparison of Mixed-Integer Programming Models for Nonconvex Piecewise Linear Cost Minimization Problems”. In: *Management Science* 49.9 (2003), pp. 1268–1273. ISSN: 00251909, 15265501. URL: http://www.jstor.org/stable/4134040.

[10] Josef Dick and Friedrich Pillichshammer. “Discrepancy theory and quasi-Monte Carlo integration”. In: *A Panorama of Discrepancy Theory*. Springer, 2014, pp. 539–619.

[11] WT Elsayed et al. “A review on accuracy issues related to solving the non-convex economic dispatch problem”. In: *Electric Power Systems Research* 141 (2016), pp. 325–332.

[12] Z.-L. Gaing. “Particle Swarm Optimization to Solving the Economic Dispatch Considering the Generator Constraints”. In: *IEEE Transactions on Power Systems* 18.3 (2003), pp. 1187–1195. DOI: https://doi.org/10.1109/TPWRS.2003.814889.

[13] Zwe-Lee Gaing. “Particle swarm optimization to solving the economic dispatch considering the generator constraints”. In: *IEEE Transactions on Power Systems* 18.3 (Aug. 2003), pp. 1187–1195. ISSN: 0885-8950. DOI: 10.1109/TPWRS.2003.814889.

[14] M. Götz. “On the Riesz energy of measures”. In: *Journal of Approximation Theory* 122.1 (2003), pp. 62–78. ISSN: 0021-9045. DOI: https://doi.org/10.1016/S0021-9045(03)00031-5. URL: http://www.sciencedirect.com/science/article/pii/S0021904503000315.

[15] Ladislav Kocis and William J Whiten. “Computational investigations of low-discrepancy sequences”. In: *ACM Transactions on Mathematical Software (TOMS)* 23.2 (1997), pp. 266–294.

[16] F. N. Lee and A. M. Breipohl. “Reserve constrained economic dispatch with prohibited operating zones”. In: *IEEE Transactions on Power Systems* 8.1 (Feb. 1993), pp. 246–254. ISSN: 0885-8950. DOI: 10.1109/59.221233.

[17] Ramon E Moore. “Global optimization to prescribed accuracy”. In: *Computers & mathematics with applications* 21.6-7 (1991), pp. 25–39.

[18] Taher Niknam, Rasoul Azizipanah-Abarghooee, and Jamshid Aghaei. “A new modified teaching-learning algorithm for reserve constrained dynamic economic dispatch”. In: *IEEE Transactions on power systems* 28.2 (2013), pp. 749–763.

[19] Raghav Prasad Parouha and Kedar Nath Das. “A novel hybrid optimizer for solving Economic Load Dispatch problem”. In: *International Journal of Electrical Power Energy Systems* 78 (2016), pp. 108–126. ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2015.11.058. URL: http://www.sciencedirect.com/science/article/pii/S0142061515004895.

[20] IEEE Report. “Present Practices in the Economic Operation of Power Systems”. Undetermined. In: *IEEE Transactions on Power Apparatus and Systems* PAS-90.4 (1971), pp. 1768–1775. DOI: 10.1109/TPAS.1971.293169.

[21] M S. P. Subathra et al. “A Hybrid With Cross-Entropy Method and Sequential Quadratic Programming to Solve Economic Load Dispatch Problem”. In: 9 (Jan. 2014), pp. 1–14.

[22] N. Sinha, R. Chakrabarti, and P. K. Chattopadhyay. “Evolutionary programming techniques for economic load dispatch”. In: *IEEE Transactions on Evolutionary Computation* 7.1 (Feb. 2003), pp. 83–94. ISSN: 1089-778X. DOI: 10.1109/TEVC.2002.806788.

[23] D.C. Walters and Gerald Sheble. “Genetic algorithm solution of economic dispatch with valve point loading”. In: 8 (Sept. 1993), pp. 1325–1332.

[24] M. Q. Wang et al. “A mixed integer quadratic programing for dynamic economic dispatch with valve point effect”. In: *IEEE Trans. Power Syst.* 29.5 (2014).

[25] Y. -. Wu and A. S. Debs. “Initialisation, decoupling, hot start, and warm start in direct nonlinear interior point algorithm for optimal power flows”. In: *IEE Proceedings - Generation, Transmission and Distribution* 148.1 (Jan. 2001), pp. 67–75. ISSN: 1350-2360. DOI: 10.1049/ip-gtd:20010008.

[26] Z.L. Wu et al. “Two-phase mixed integer programming for non-convex economic dispatch problem with spinning reserve constraints”. In: *Electric Power Systems Research* 140 (2016), pp. 653–662. ISSN: 0378-7796. DOI: https://doi.org/10.1016/j.epsr.2016.05.006. URL: http://www.sciencedirect.com/science/article/pii/S0378779616301584.

[27] X Xia and AM Elaiw. “Dynamic economic dispatch: A review”. In: *Online Journal on Electronics and Electrical Engineering (OJEEE)* 2.2 (2010).

Appendix A

Appendices

A.1 Abbreviations

Here is the list of the abbreviations used in the thesis.

- MIQP : mixed-integer quadratic program
- APQUA : Adaptative Piecewise Quadratic UnderApproximation
- QCQP: Quadrtically constrained quadratic programming
- RLT : Relaxation linearization technique
- VPE : Valve-point effect
- UCL : Université Catholique de Louvain

A.2 Variables and parameters

Here are a list of the variables and a list of the coefficients present in the work.

A.2.1 Variables

p_i	MW	production of the reactor i
$p_{i,t}$	MW	production of the reactor i for the hour t
$p_{i,j}$	MW	production of the reactor i for the fuel j
$\chi_{i,j}$	MW	production of the reactor i on the linear spline j
$\eta_{i,j}$		binary variable representing whether the reactor i operatees in the jth segment of the piecewise linear cost curve
$g(p)$	\$	piecewise linearized cost
$f(p)$	\$	cost
$SR_{i,t}$	MW	spinning reserve for the reactor i at hour t
$Y_{i,j}$	MW^2	product of $p_i p_j$
$u_{i,j}$		binary variable representing whether the reactor i uses the fuel j
$\xi_{i,j}$		binary variable representing whether the reactor i operates in the operating zone j

A.2.2 Parameters

a_i	$\$/MW^2$	coefficient of the quadratic function for reactor i
b_i	$\$/MW$	coefficient of the quadratic function for reactor i
c_i	\$	coefficient of the quadratic function for reactor i
d_i	$\$/MW$	coefficient of the valve-point effect function for reactor i
e_i		coefficient of the valve-point effect function for reactor i
D	MW	demand of the system
p_i^{min}	MW	minimum production of a reactor i
p_i^{max}	MW	maximum production of a reactor i
n		number of units
n_i^{kink}		number of kink points for unit i
m		number of hours
n_i^{knot}		number of segment on piecewise linearized curve for the reactor i
n_i^{fuel}		number of fuel for unit i
$\alpha_{i,j}$	$\$/MW$	cost coefficient for the piecewise linearized curve, refers to the jth segment and the unit i
$\beta_{i,j}$	\$	cost coefficient for the piecewise linearized curve, refers to the jth segment and the unit i
UR_i	MW	upper ramp rate limit for unit i
DR_i	MW	lower ramp rate limit for unit i
SSR_t^{req}	MW	required spinning reserve at time t
$B_{i,j}$	$\$/MW^2$	element in the system loss coefficient matrix B
B_{0j}	$\$/MW^2$	element in the system loss coefficient vector B
B_{00}	$\$/MW^2$	element in the system loss coefficient term B
$\mathcal{S}$		ellipsoid created by the load demand with loss term
M		number of planes in the low discrepancy sequences used to generate the plane approximation
$p_{i,j}^{OL}$		minimum power of the feasible operating zone j for the unit i
$p_{i,j}^{OU}$		maximum power of the feasible operating zone j for the unit i

A.3 Transformation from MVA base unit coefficients to per unit value

The MVA base is the basis for the mega volt-ampere. The volt-ampere is a unit that measures the apparent power of an electric circuit. It is equal to the product of root-mean-square (RMS) voltage and RMS current. In direct current (DC) circuits, this product is equal to the real power (active power) in watts. Volt-amperes are useful only in the context of alternating current (AC) circuits (sinusoidal voltages and currents of the same frequency).

Two main methods are used to switch from a coefficient in VA to per unit.[11] The first idea is to employ the actual values of the loss coefficients that are obtained in the following equation :

$$B_{i,j} = \frac{B_{i,jp.u}}{\text{MVAbase}} \qquad B_{0i} = B_{0ip.u} \qquad B_{00} = B_{00p.u}\text{MVAbase} \tag{A.1}$$

where $B_{i,jp.u}$, $B_{0ip.u}$, B_{00} are the per unit values of the loss coefficients. MVA base is the base MVA value. After, the loss coefficients returned from (A.1) are introduced in (4.2) with the power generation value obtained by the tested algorithm in MW. So, the transmission losses are also in MW, which is the actual value of those losses.

The second approach is to transform the MW creation value obtained by the observed algortihm to per unit values using the equation:

$$p_{ip.u} = \frac{p_i}{MVAbase} \tag{A.2}$$

where $p_{ip.u}$ is the per unit value production of the reactor i. The output from (A.2) is replaced by the given per-unit loss coefficients in (4.2) to calculate the total transmission losses and which will be a per-unit value ($p_{Lp.u}$). The actual transmission losses p_l are obtained as:

$$p_L = p_{Lp.u}\text{MVAbase} \ .$$

Both strategies returned the same transmission losses.

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