A new MILP-based approach for Unit Commitment in power production planning.

Ana Viana and João Pedro Pedroso

Abstract—**This paper presents a novel, iterative optimisation algorithm for thermal unit commitment in power generation planning. The algorithm, based on a mixed-integer formulation of the problem, considers piecewise linear approximations of the quadratic fuel cost function that are dynamically updated in an iterative way, converging to the optimum.**

From extensive computational tests in a broad set of benchmark instances of this problem, the algorithm is found to be flexible and capable of easily incorporating different problem constraints. Moreover, it can solve large problems.

Most importantly, optimal solutions were obtained for several well-known benchmark instances that are not known to have been solved to optimality before.

Index Terms—**Unit Commitment, Combinatorial Optimisation, Mixed-integer Programming.**

NOTATION

Constants

- T – length of the planning horizon.
- $\mathcal{T} = \{1, \ldots, T\}$ – set of planning periods.
- $\mathcal{U}$ – set of units.
- $P_u^{\min}, P_u^{\max}$ – minimum and maximum production levels of unit u.
- $T_u^{\text{on}}, T_u^{\text{off}}$ – minimum number of periods unit u must be kept switched on/off.
- $r_u^{\text{up}}, r_u^{\text{down}}$ – maximum up/down rates of unit u.
- D_t – system load requirements in period t.
- R_t – spinning reserve requirements in period t.
- a_u, b_u, c_u – fuel cost parameters for unit u.
- $a_u^{\text{hot}}, a_u^{\text{cold}}$ – hot and cold start up costs for unit u.
- t_u^{cold} – number of periods after which start up of unit u is evaluated as cold.
- y_u^{prev} – previous state of unit u (1 if on, 0 if off).
- t_u^{prev} – number of periods unit u has been on or off prior to the first period of the planning horizon.

Variables

- Decision variables:
  - y_{ut} – 1 if unit u is on in period t, 0 otherwise.
  - p_{ut} – production level of unit u, in period t.
- Auxiliary variables[1]:
  - $x_{ut}^{\text{on}}, x_{ut}^{\text{off}}$ – 1 if unit u is started/switched off in period t, 0 otherwise;
  - s_{ut}^{hot} – 1 if unit u has a hot start in period t, 0 otherwise;
  - s_{ut}^{cold} – 1 if unit u has a cold start in period t, 0 otherwise;
  - $p_{ut}^{\max}$ – maximum production levels of unit u in period t (due to ramp constraints).
- Production costs
  - $F(p_{ut})$ – fuel cost of unit u in period t.
  - $S(x_{ut}^{\text{off}}, y_{ut})$ – start up cost of unit u in period t.
  - H_{ut} – shut down cost of unit u in period t.

Ana Viana is with the School of Engineering, Polytechnic Institute of Porto, Portugal, e-mail: aviana@inescporto.pt.

João Pedro Pedroso is with the Department of Computer Science, Faculty of Sciences, University of Porto, Portugal, e-mail: jpp@fc.up.pt.

João Pedro Pedroso and Ana Viana are with INESC Porto.

[1]Due to model structure, some of the binary variables can be relaxed to the set $[0, 1]$, as discussed later.

I. INTRODUCTION

THE Unit Commitment problem (UCP) is the problem of deciding which power generator units must be committed/decommitted over a planning horizon (that lasts from 1 day to 2 weeks, generally split into periods of 1 hour each). The production levels at which units must operate (pre-dispatch) must also be determined for optimising a given objective function, and the committed units must generally satisfy the forecasted system load and reserve requirements, as well as a large set of technological constraints.

This problem has major practical significance because the effectiveness of the schedules obtained has strong economical impact for any power generation company. Due to this reason and to its high complexity, it has received considerable research attention. Even after several decades of intensive study, it is still a rich and challenging topic of research.

Proposed optimisation techniques for unit commitment encompass very different paradigms, ranging from exact approaches and Lagrangian relaxation to some rule of thumb, or very elaborate heuristics and metaheuristics. In the past, the combinatorial nature of the problem and its multi-period characteristics prevented exact approaches from being successful in practice: they resulted in very inefficient algorithms that were only capable of solving small problem instances of virtually no practical interest. Heuristic techniques, as those based in priority lists, did not totally succeed neither, as they tend to lead to low quality solutions. Concerning meta-heuristics, they had very promising outcomes when they were first explored. The quality of the results was better than the ones achieved by well established techniques, and good solutions were obtained very quickly. Some drawbacks can however be pointed out when metaheuristics are used. One major drawback, if one considers that the ultimate goal is to design techniques that can be accepted and used by a company, is the dependence of metaheuristics on parameter tuning. Tuning the parameters is time consuming and the complex tuning procedure requires deep knowledge on the algorithm implemented. Furthermore, accurate tuning is vital for good algorithm performance. A second drawback has to do with the lack of information metaheuristics provide in terms of solution quality (i.e., how far it is from the optimal solution). Some proposals have been made to soften the referred drawbacks; but this still remains an open line of research.

Currently, the dramatic increase in efficiency of mixed-integer programming (MIP) solvers encourages thorough exploitation of their capabilities. Some research has already been directed towards the definition of alternative, more efficient, mixed-integer linear programming (MILP) formulations of the problem (see [1], [2]). Extensive surveys of different optimisation techniques and modelling issues are provided in [3] and [4].

This paper proposes a MIP formulation for quadratic optimisation of the UCP, and also presents a method based on a new linear formulation, which shows to be effective at solving problems of relevant practical size. Instead of considering a quadratic representation of the fuel cost, the linear model considers a piecewise linear approximation of the function and, in an iterative process, updates it by including additional pieces. Function updating is based on the solutions obtained in the previous iteration.

The solution approach developed in this research was tested on several well-known test instances that are not known to have been solved to optimality before. For each of them, the new approach iteratively converged to the optimal solution, even for the largest benchmark instances.

II. Problem variants

Different modelling alternatives, reflecting different problem issues such as fuel, multiarea and emission constraints have been published (e.g. [5], [6], [7]). More recently, security constraints [8] and market related aspects [9] have been addressed.

The decentralised management of production has also brought up new issues to the area [10] and in some markets the problem is now reduced to single-unit optimisation. However, for several decentralised markets the traditional problem is still very much similar to that of the centralised markets [1], [2]; the main difference is the objective function that, rather than minimising production costs, maximises total welfare. Therefore, the techniques that apply for centralised management of production will also be effective at solving many decentralised market production problems.

This paper considers the centralised UCP model. The objective of the problem is to minimise total production costs over a given planning horizon. The total production cost is expressed as the sum of fuel costs (quadratic functions that depend on the production level of each unit) and start-up costs. Start-up costs are represented by constants that depend on the last period the unit was operating. Besides uninterrupted operation of the unit (i.e., no start-up cost[2]), two constants are defined: one constant for hot start-up costs, when the unit has been off for a number of periods smaller or equal to a given value, and the other for cold start-up costs.

The following constraints will be included in the formulation: system power balance demand, system reserve requirements, unit initial conditions, unit minimum up and down times, generation limits and ramp constraints. For a standard quadratic mathematical formulation readers can refer to [11].

III. MILP formulations for the UCP

For many years, solution approaches for the UCP were mainly based on Lagrangian relaxation and (meta)heuristics. This was due to the non-existence of exact approaches capable of coping with the computational complexity of the problem within reasonable resources. However, dramatic improvement of MIP solvers in recent years suggested that an effort should be applied to study "good" mathematical formulations of the problem, so that it can be handled by relevant solvers.

The first requirement is the linearisation of the many non-linearities in the problem; namely, minimum up and down constraints, minimum and maximum power production constraints (for problems that consider ramps), and the objective function.

Several efforts have been made to improve and strengthen the formulation of the UCP; pioneering work can be found in [12]. This work considers three sets of binary variables that model, namely, the state of each unit, start-ups and shut-downs. The quadratic fuel cost function is represented by a piecewise-linear cost function. Initial attempts to solve the problem with standard branch-and-bound (B&B) proved to be inefficient. As a result, an extended version of the algorithm that considers problem-specific characteristics in the branching process was proposed. Results are provided for problems up to 16 units and 14 time periods.

A thorough discussion on model linearisation, considering a perfect electricity spot market and a single unit (self-scheduling), is given in [13]. The quadratic cost function is approximated by a piecewise linear cost function with L segments. Three extra sets of variables are required when compared to the model in [12]: 0/1 variables equal to 1 if the unit is started-up at the beginning of hour t and it has been off for k hours; integer variables to set the number of hours the unit has been on or off at the end of hour t; and, 0/1 variables to equal to 1 if the power output of the unit at hour t exceeds segment l of the piecewise linear approximation of the quadratic cost function.

The authors of [14] focus their study on an accurate modelling of start-up and shut-down power trajectories (that depend on ramps). Similar to [13], [15] presents a model that can adapt to centralised or competitive markets. The model is validated with a 27 unit $\times$ 24 hour instance, but no information on the efficiency of the algorithm is reported. Later, a model introduced in [1] reduces the number of binary variables as well as the number of constraints of previous formulations. Wu presents in [16] a segment partition methodology to determine segment partition points that shall provide a tighter piecewise linear approximation of the quadratic cost function. In such

[2]Notice that even in this situation there is a fixed component in the quadratic cost function.

case the quadratic cost function is still approximated by a piecewise linear cost function with L segments, but each segment may have a different length.

Still within the scope of MILP formulations for the UCP, a study on the quality of previously proposed valid inequalities is made in [17]. In particular, the authors focus on valid inequalities used for minimum up and down time constraints and show how some inequalities can be improved. Other work in this area can be found in [18]. It concludes that by adding "perspective cuts" (a family of valid inequalities) to the problem, effectiveness and efficiency of the solvers improve. The same authors (together with Lacalandra) proposed a MILP formulation in [2] in which the "perspective cuts" concept is used to approximate the nonlinear objective function.

IV. New MILP formulation

Although quality of MILP solvers has improved dramatically in recent years, mathematical models such as the one given in [11] are not suitable for those solvers due to their many nonlinearities; to fully utilise MILP solvers, these nonlinearities must be removed from the model, if possible. In the following subsections we present and discuss a linearised mathematical model for the UCP. The model was implemented in AMPL (*A Modelling Language for Mathematical Programming*) [19], and the CPLEX MIP solver was used for its solution.

A. System constraints

Two types of system constraints are considered: system power balance (Constraints (1)) and spinning reserve requirements (Constraints (2)). The impact of ramp constraints can be considered when setting reserve constraints; this can be achieved by using variable $p^{\max}_{ut}$, rather than the constant $P^{\max}_{u}$, in Equation (2).

$$\sum_{u \in \mathcal{U}} p_{ut} = D_t, \qquad \forall t \in \mathcal{T}, \tag{1}$$

$$\sum_{u \in \mathcal{U}} p^{\max}_{ut} \geq D_t + R_t, \quad \forall t \in \mathcal{T}, \tag{2}$$

with:

$$\begin{aligned} p^{\max}_{ut} &\leq y_{ut} P^{\max}_u, \qquad \forall u \in \mathcal{U}, \text{ for } t = 2 \ldots T, \\ p^{\max}_{ut} &\leq p_{u,t-1} + y_{u,t-1} r^{\text{up}}_u + P^{\max}_u (1 - y_{u,t-1}), \\ &\qquad \forall u \in \mathcal{U}, \text{ for } t = 2 \ldots T. \end{aligned}$$

B. Technical constraints

Technical constraints represent limitations of the generating units and constrain the overall system performance (e.g., units minimum up and down times, production limits, and ramps).

1) Minimum up and down times: When a unit u is switched on (off), it must remain on (off) for at least T^{on}_u (T^{off}_u) consecutive periods. Constraints (3) and (4) model this aspect for the initial state, while constraints (5) and (6) do the same for the remaining planning horizon. In Equation (3) θ^{on}_u represents $\max(0, T^{\text{on}}_u - t^{\text{prev}}_u)$, and θ^{off}_u in Equation (4) stands for $\max(0, T^{\text{off}}_u - t^{\text{prev}}_u)$.

$$y_{ut} = 1, \quad \forall u \in \mathcal{U} : y^{\text{prev}}_u = 1, \text{ for } t = 0, \ldots, \theta^{\text{on}}_u, \tag{3}$$

$$y_{ut} = 0, \quad \forall u \in \mathcal{U} : y^{\text{prev}}_u = 0, \text{ for } t = 0, \ldots, \theta^{\text{off}}_u. \tag{4}$$

In Equations (5) and (6), τ^{on}_{ut} and τ^{off}_{ut} stand for $\max(t - T^{\text{on}}_u + 1, 1)$ and $\max(t - T^{\text{off}}_u + 1, 1)$, respectively.

$$\sum_{i=\tau^{\text{on}}_{ut}}^{t} x^{\text{on}}_{ui} \leq y_{ut}, \qquad \forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \tag{5}$$

$$\sum_{i=\tau^{\text{off}}_{ut}}^{t} x^{\text{off}}_{ui} \leq 1 - y_{ut}, \quad \forall u \in \mathcal{U}, \forall t \in \mathcal{T}. \tag{6}$$

2) Generation limits and ramps: Power production levels of thermal power units are within a range defined by the technical minimum and maximum production levels in Constraints (7).

$$P^{\min}_u y_{ut} \leq p_{ut} \leq P^{\max}_u y_{ut}, \quad \forall u \in \mathcal{U}, \forall t \in \mathcal{T}. \tag{7}$$

If ramps are considered, i.e., if the difference of values in production levels is limited to a maximum value in consecutive periods, additional constraints are needed. Constraints (8) and (9) model, respectively, maximum up and down rates for each unit in consecutive periods of time.

$$p_{ut} - p_{u,t-1} \leq r^{\text{up}}_u, \qquad \forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \tag{8}$$

$$p_{u,t-1} - p_{ut} \leq r^{\text{down}}_u, \quad \forall u \in \mathcal{U}, \forall t \in \mathcal{T}. \tag{9}$$

C. Additional constrains

A set of additional constraints that allow the computation of auxiliary variables also allows relaxation of integrality for variables x^{on}_{ut} and x^{off}_{ut}, as discussed below.

1) Setting and computation of variables s^{hot}_{ut} and s^{cold}_{ut}: Constraints (10) states that every time a unit is switched on, a start-up cost will be incurred.

$$s^{\text{hot}}_{ut} + s^{\text{cold}}_{ut} = x^{\text{on}}_{ut}, \quad \forall u \in \mathcal{U}, \forall t \in \mathcal{T}. \tag{10}$$

Constraints (11) determine the start cost of each unit, i.e., decide whether it is a cold or a hot start cost. It will be a cold start cost if the unit remained off for t^{cold}_u periods of time, or more; and a hot start cost otherwise.

$$y_{ut} - \sum_{i=t-t^{\text{cold}}_u-1}^{t-1} y_{ui} \leq s^{\text{cold}}_{ut}, \quad \forall u \in \mathcal{U}, \forall t \in \mathcal{T}. \tag{11}$$

2) Setting and computation of variables x^{on}_{ut} and x^{off}_{ut}: Constraints (12) determine each unit's switch-on variables, and Constraints (13) determine the switch-off variables.

$$y_{ut} - y_{u,t-1} \leq x^{\text{on}}_{ut}, \qquad \forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \tag{12}$$

$$x^{\text{off}}_{ut} = x^{\text{on}}_{ut} + y_{u,t-1} - y_{ut}, \quad \forall u \in \mathcal{U}, \forall t \in \mathcal{T}. \tag{13}$$

3) Relaxation of integrality constraints on variables x^{on}_{ut} and x^{off}_{ut}: Constraints (10) and (13) allow relaxation of variables x^{on}_{ut} and x^{off}_{ut}. In fact, if s^{hot}_{ut} and s^{cold}_{ut} are defined as binary variables, through constraint (10) x^{on}_{ut} will always be 0 or 1. Furthermore, through constraints (13), since y_{ut} is binary, x^{off}_{ut} will always be set to 0 or 1, for feasible y_{ut}.

D. Objective function

The objective of this problem is to minimise the total production cost over the planning horizon, expressed as the sum of fuel, start-up and shut-down costs (Equation (14)).

$$\text{minimize} \sum_{t\in\mathcal{T}} \sum_{u\in\mathcal{U}} \left(F(p_{ut}) + S(x^{\text{off}}_{ut}, y_{ut}) + H_{ut} \right). \quad (14)$$

We consider the traditional quadratic function for $F(p_{ut})$, as follows:

$$F(p_{ut}) = \begin{cases} c_u p_{ut}^2 + b_u p_{ut} + a_u & \text{if } y_{ut} = 1, \\ 0 & \text{otherwise.} \end{cases} \quad (15)$$

Shut-down costs will be set to zero and start-up costs are modelled as in [20]:

$$S\left(x^{\text{off}}_{ut}, y_{ut}\right) = y_{ut}(1 - y_{u,(t-1)}) S_x(x^{\text{off}}_{ut}). \quad (16)$$

S_x depends on the last period the unit was operating as follows:

$$S_x = \begin{cases} a^{\text{hot}}_u & \text{if } x^{\text{off}}_{ut} \le t^{\text{cold}}_u, \\ a^{\text{cold}}_u & \text{otherwise.} \end{cases} \quad (17)$$

In this work, the following linearised function (equivalent to the one in [20]) is used to represent start-up-costs:

$$S\left(x^{\text{off}}_{ut}, y_{ut}\right) = a^{hot}_u s^{hot}_{ut} + a^{cold}_u s^{cold}_{ut}. \quad (18)$$

V. Iterative linear algorithm

The new solution approach considers a piecewise linear approximation of the quadratic fuel cost function (see Equation (15)), where a linear MILP model is iteratively solved. The MILP will provide increasing precision at each iteration, until reaching a user-defined proximity to the quadratic function.

As the cost function is convex, if we find a linear function tangent to it, and constrain the cost to be greater than the value of the linear function, we have a lower approximation of the cost. The process proposed is to dynamically find linear functions, tangent to the true cost at points where it is being underestimated, and add them to a set. We then impose that the cost of any production level p must be greater than the maximum of those linear functions, evaluated at p.

For clarity, let us remove the indices u, t identifying the generator and time period, respectively. For any generator and any period, we start by approximating its cost by means of two linear functions: one going through $(P^{\min}, F(P^{\min}))$, and another going through $(P^{\max}, F(P^{\max}))$, as shown in Figure 1.

After solving the problem using this approximation, we obtain a production level for a unit of, say, p. The operating cost at this point will be underestimated as the value of the highest of the straight lines at p; in Figure 1 this is given by the value $\tilde{F}$. In order to exclude this point from the feasible region, we add another linear function to our set: the line tangent to the quadratic function, evaluated at p, and represented in blue in Figure 2. When we solve the problem with this additional constraint added, the solution may change; the optimal production level for this same unit may now be another possible value p', as shown in Figure 2. As we add more and more tangents and select the highest, we converge to an exact representation of the true cost function.

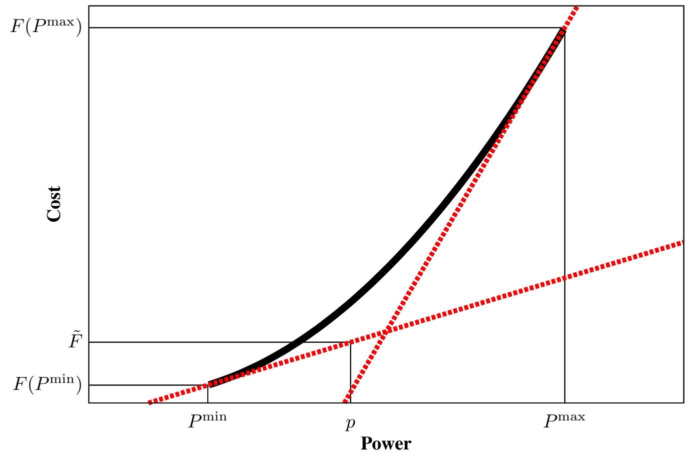

Fig. 1. *Initial approximation of the cost function by two straight lines, going through the minimum and maximum operating power of the unit. If the current production level for this unit is p, its cost (in this iteration) will be approximated by F*

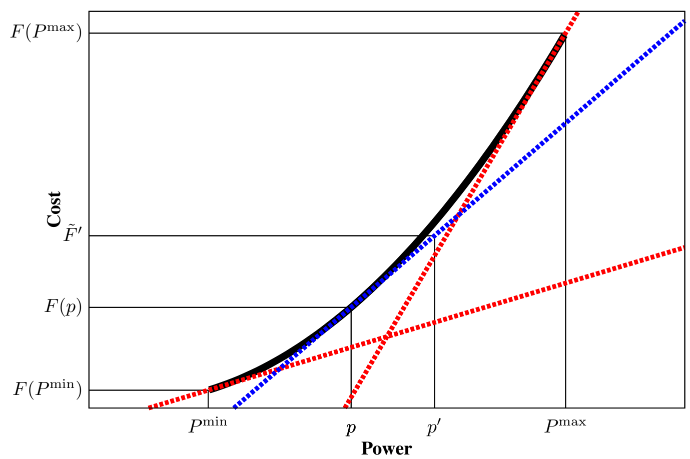

Fig. 2. *Approximation of the cost function by the maximum of three straight lines, after obtaining production at level p on the previous iteration.*

A. Algorithm description

For each unit, we start with the corresponding quadratic fuel cost function $F(p)$ approximated by two linear functions, the first being tangent to $F(p)$ at the minimum power $(P^{\min}, F(P^{\min}))$, and the second being tangent at $(P^{\max}, F(P^{\max}))$ (as shown in Figure 1).

Thereafter, more straight lines are iteratively added into a set, until having one iteration with all production levels being correctly evaluated, up to an acceptable error.

Let $\mathcal{P}$ be a set of numbers identifying the power at which new tangents to the true cost are added; initially $\mathcal{P} = \{P^{\min}, P^{\max}\}$. At a given iteration, let the production level obtained in the MILP solution be p, and the corresponding cost approximation (i.e., the maximum of the linear cost functions evaluated at p) be $\tilde{F}$. We add the point p to the set $\mathcal{P}$ whenever $|F(p) - \tilde{F}|/F(p) > \epsilon$, were ϵ is a user-defined tolerance. Otherwise, we accept the current approximation as accurate enough.

In the MILP solved at each iteration, we add the following constraints (making sure that they are only imposed if the corresponding unit is switched on at the period considered):

$$F_{ut} \geq \alpha_{un} + \beta_{un}(p_{ut} - \bar{p}_n), \quad \text{for } n = 1, \ldots, |\mathcal{P}|,$$

now using the actual variables p_{ut} for production level and F_{ut} for production cost, of a given unit at a given period. For a given unit, and for each production level $\bar{p}_n$ where the approximation does not satisfy the tolerance, the constants of the above straight lines are obtained by:

$$\alpha_{un} = c_u \bar{p}_n^2 + b_u \bar{p}_n + a_u,$$
$$\beta_{un} = 2c_u \bar{p}_n + b_u.$$

The algorithm stops when in a given iteration the set $\mathcal{P}$ is unchanged, i.e., no straight lines are added, meaning that the production costs are all already being correctly evaluated up to the specified tolerance ϵ. In our experiment, we have set $\epsilon = 10^{-6}$; this allows an excellent approximation of the quadratic function in all the instances tested (actually, we could observe no difference between the quadratic function and the linear approximation, concerning the solutions obtained).

VI. Computational results

The algorithm was tested in two sets of instances: one without ramp constraints but that has for long been a reference for comparison of UC algorithms [20] (instances P1 through P6); and the other with ramp constraints (instances R1 through R6). CPU times were obtained with CPLEX 12.1 on a computer with Quad-Core Intel Xeon processor at 2.66 GHz and running Mac OS X 10.6.6; only one core was assigned to this experiment.

Tables I and II present the results obtained using the algorithm proposed in this paper for different sets of UCP instances. Instances P1 to P6, in Table I, are the standard Kazarlis [20] benchmarks, which do not include ramp constraints. Ramp constraints are considered in instances R1 to R6 (Table II), resulting from instances P1 to P6, by setting ramp up-and-down maximum values identical to the minimum production level of each unit. All instances consider a 24-hour planning horizon, with one period per hour, and the number of units ranged from 10 to 100. Empty entries in the tables mean that the solver could not find the solution within 24 hours of CPU time.

Table III presents the results obtained using the algorithm proposed in this paper for instances P1 to P6 when start-up costs are evaluated by Equation (19). This equation was first proposed in [21] and was later used by several authors.

$$S_x = \begin{cases} a_u^{\text{hot}} & \text{if } T_u^{\text{off}} < x_{ut}^{\text{off}} \leq T_u^{\text{off}} + t_u^{\text{cold}}, \\ a_u^{\text{cold}} & \text{otherwise.} \end{cases} \tag{19}$$

Table IV presents results reported in the literature for instances P1 to P6, using different heuristic methods. Although the objective function value reported in this paper (565828) for the 10 unit instance using the iterative linear algorithm is higher than the one reported in some papers (565825), the actual solution is the same. Small differences in values can be attributed to possible rounding of values by other authors. Similarly, Table V presents results reported in the literature for instances P1 to P6, when start-up costs are modelled by Equation (19). Naturally, the results reported in Tables IV and V cannot be compared. At most, the results in Table V can be seen as lower bounds for instances P1 to P6.

TABLE I
Results for instances P1 to P6 (standard Kazarlis set).

		Iterative linear alg.		Quadratic model	
Instance	Size	Objective	CPU	Objective	CPU
P1	10	565 828	0.30	565 828	1.95
P2	20	1 126 000	6.16	1 126 000	241.
P3	40	2 248 280	58.5	2 248 280	22716.
P4	60	3 368 950	541.		
P5	80	4 492 170	19562.		
P6	100	5 612 690	11111.		

TABLE II
Results for insctances R1 to R6 (including ramp constraints).

		Iterative linear alg.		Quadratic model	
Instance	Size	Objective	CPU	Objective	CPU
R1	10	570 396	3.01	570 396	5.06
R2	20	1 135 450	222.	1 135 450	1741.
R3	40	2 267 540	7305.		
R4	60	3 398 610	3329.		
R5	80	4 531 720	60374.		
R6	100	5 662 900	104152.		

In Tables I and II, columns *Quadratic model* provide the optimal result for the base problem and columns *Iterative linear algorithm* provide the results obtained by the method we propose, confirming that the algorithm converges to the optimal solution. Columns CPU refer to the time spent (in seconds) to solve the problem by each of the methods. Attempts to solve the problem with the quadratic formulation without ramps were not successful for instances with more than 40 units; for the problem with ramps, no success was obtained with the quadratic formulation for instances with more than 20 units.

To the best of our knowledge, no optimal solutions have ever been reported for instances P1 to P6, with quadratic cost function, even for the smallest one. We have shown that for problems up to 40 units, optimal solutions can be obtained by current, state-of-the-art MIP solvers. Moreover, with the iterative approach based on the linear approximative model, we were able to reach the optimal solution with dramatic reductions in CPU times, when compared to direct solution approach with the quadratic solver of CPLEX, having determined the optimal solutions (with a tolerance $\epsilon = 10^{-6}$ on production costs) for all the instances. These can be compared to the best published values for the quadratic models (see Tables IV and V).

Similar conclusions may be made for the ramp problem. The quadratic solver of CPLEX was capable of reaching optimal solutions for problem instances of up to 20 units. Optimal values for the whole set of problems were, again, reached by the iterative linear algorithm, albeit using larger CPU times.

TABLE III
RESULTS FOR PROBLEMS P1 TO P6 (MODIFIED COST FUNCTION).

		Iterative linear alg.		Quadratic model	
Instance	Size	Objective	CPU	Objective	CPU
P1	10	563 938	0.55	563 938	1.69
P2	20	1 123 300	12.8	1 123 300	142.
P3	40	2 242 580	978.		
P4	60	3 359 950	309.		
P5	80				
P6	100	5 597 770	4600.		

TABLE IV
PREVIOUS RESULTS FOR PROBLEMS P1 TO P6: BEST SOLUTION FOUND.

Instance	Size	LR [20]	GA [20]	LR–MA [22]
P1	10	565 825	565 825	565 827
P2	20	1 130 660	1 126 243	1 127 254
P3	40	2 258 503	2 251 911	2 249 589
P4	60	3 394 066	3 376 625	3 370 595
P5	80	4 526 022	4 504 933	4 494 214
P6	100	5 657 277	5 627 437	5 616 314
Instance	Size	ICGA [23]	GRASP [11]	CON [24]
P1	10	566 404	565 825	565 825
P2	20	1 127 244	1 126 805	1 126 070
P3	40	2 254 123	2 255 416	2 248 490
P4	60	3 378 108	3 383 184	3 370 530
P5	80	4 498 943	4 524 207	4 494 140
P6	100	5 630 838	5 668 870	5 615 410

TABLE V
PREVIOUS RESULTS FOR PROBLEMS P1 TO P6: BEST SOLUTION FOUND, USING THE MODIFIED COST FUNCTION.

Instance	Size	MILP-UC [1]	QBPSO [25]	QEA-UC [26]
P1	10		539 977	563 938
P2	20		1 123 297	1 123 607
P3	40		2 242 957	2 245 557
P4	60		3 361 980	3 366 676
P5	80		4 482 085	4 488 470
P6	100	5 605 189	5 602 486	5 609 550
Instance	Size	IQEA-UC* [27]	SFLA [28]	ICA [29]
P1	10	563 938	564 769	563 938
P2	20	1 123 297	1 123 261	1 124 274
P3	40	2 242 980	2 246 005	2 247 078
P4	60	3 362 010	3 368 257	3 371 722
P5	80	4 482 826	4 503 928	4 497 919
P6	100	5 602 387	5 624 526	5 617 913

VII. CONCLUSIONS AND FURTHER DEVELOPMENTS

The main contribution of this paper is a novel, efficient methodology for approximating the quadratic cost of electricity generating units, with an iterative method that uses of a linear model, converging to the exact solution. The paper also establishes optimal solutions for small instances of the quadratic model (with and without ramp constraints) showing that current, state-of-the-art solvers can tackle problems that were not solvable before.

Computational analysis shows that the method is capable of reaching the optimum of the quadratic model, when this is known, using much less computational time than required by its quadratic programming solution. For large problem instances, where the quadratic model could not be solved directly in a reasonable time, the iterative linear algorithm has found the optimal solution.

Similar conclusions can be made when ramp constraints are considered. The iterative linear algorithm is also capable of reaching the optimum to all the instances of the quadratic model. This is particularly important for instances with more than 20 units, for which straightforward use of the quadratic programming solver was not successful.

As future work, we plan to extend the algorithm in order to allow its use for the solution of extended models, e.g. including other electricity generation technologies in addition to thermal units. The approach proposed for approximating the quadratic cost can be applied to any convex function; in terms of practical applications, this is a feature that deserves being explored, as it may allow a better (instead of quadratic) model of the true cost function.

ACKNOWLEDGEMENTS

Financial support for this work was provided by the Portuguese Foundation for Science and Technology (under Project PTDC/EGE-GES/099120/2008) through the "Programa Operacional Temático Factores de Competitividade (COMPETE)" of the "Quadro Comunitário de Apoio III", partially funded by FEDER.

REFERENCES

[1] M. Carrion and J. Arroyo, "A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem," *IEEE Transactions in Power Systems*, vol. 21, no. 3, pp. 1371–1378, 2006.
[2] A. Frangioni, C. Gentile, and F. Lacalandra, "Tighter approximated milp formulations for unit commitment problems," *Power Systems, IEEE Transactions on*, vol. 24, no. 1, pp. 105 –113, Feb. 2009.
[3] N. Padhy, "Unit commitment – a bibliographical survey," *IEEE Transactions in Power Systems*, vol. 19, no. 2, pp. 1196–1205, 2004.
[4] H. Yamin, "Review on methods of generation scheduling in electric power systems," *Electric Power Systems Research*, vol. 69, pp. 227–248, 2004.
[5] F. Lee, "A fuel constrained unit commitment method," *IEEE Transactions on Power Systems*, vol. 4, pp. 1208–1218, 1989.
[6] Z. Ouyang and S. Shahidehpour, "Heuristic multi-area unit commitment with economic dispatch," *IEE Proceedings – C*, vol. 138, pp. 242–252, 1991.
[7] D. Srinivasan and A. Tettamanzi, "An evolutionary algorithm for evaluation of emission compliance options in view of the clean air act amendments," *IEEE Transactions on Power Systems*, vol. 12, no. 1, pp. 336–341, 1997.
[8] Y. Fu and M. Shahidehpour, "Fast scuc for large-scale power systems," *Power Systems, IEEE Transactions on*, vol. 22, no. 4, pp. 2144 –2151, Nov. 2007.
[9] J. Xu and R. Christie, "Decentralised unit commitment in competitive energy markets," in *The Next Generation of Electric Power Unit Commitment Models*, B. Hobbs, M. Rothkopf, R. O'Neill, and H. Chao, Eds. Kluwer Academic Publishers, 2001, pp. 293–315.
[10] B. Hobbs, M. Rothkopf, R. O'Neill, and H. Chao, Eds., *The Next Generation of Electric Power Unit Commitment Models*. Kluwer Academic Publishers, 2001.
[11] A. Viana, J. Sousa, and M. Matos, "Using GRASP to solve the unit commitment problem," *Annals of Operations Research*, vol. 120, no. 1, pp. 117–132, 2003.
[12] T. Dillon, K. Edwin, H.-D. Kochs, and R. Taud, "Integer programming approach to the problem of optimal unit commitment with probabilistic reserve determination," *IEEE Transactions on Power Apparatus and System*, vol. vol. PAS-97, no. 6, pp. 2154–2166, 1978.
[13] J. M. Arroyo and A. J. Conejo, "Optimal response of a thermal unit to an electricity spot market," *IEEE Transactions on Power Systems*, vol. 15, no. 3, pp. 1098–1104, 2000.
[14] A. J. M. and A. J. Conejo, "Modelling of start-up and shut-down power trajectories of thermal units," *IEEE Transactions on Power Systems*, vol. 19, pp. 1562–1568, 2004.
[15] G. Chang, Y. Tsai, C. Lai, and J. Chung, "A practical mixed integer linear programming based approach for unit commitment," in *Proceedings of the 2004 IEEE Power Engineering Society General Meeting*, vol. Vol.1, Toronto, Canada, 2004, pp. 221 – 225.

[16] L. Wu, "A tighter piecewise linear approximation of quadratic cost curves for unit commitment problems," *Power Systems, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2011.
[17] K. Hedman, R. O'Neill, and S. Oren, "Analyzing valid inequalities of the generation unit commitment problem," in *Proceedings of 2009 IEEE Power System Conference and Exposition (PSCE)*, Seattle, USA, 2009.
[18] A. Frangioni and C. Gentile, "Perspective cuts for a class of convex 0-1 mixed integer programs," *Mathematical Programming*, vol. 106, pp. 225–236, 2009.
[19] "AMPL website," 2011, available: http://www.ampl.com/.
[20] S. Kazarlis, A. Bakirtzis, and V. Petridis, "A Genetic Algorithm solution to the unit commitment problem," *IEEE Transactions on Power Systems*, vol. 11, pp. 83–92, 1996.
[21] K. Juste, H. Kita, E. Tanaka, and J. Hasegawa, "An Evolutionary Programming solution to the unit commitment problem," *IEEE Transactions on Power Systems*, vol. 14, pp. 1452–1459, 1999.
[22] J. Valenzuela and A. Smith, "A seeded memetic algorithm for large unit commitment problems," *Journal of Heuristics*, vol. 8, no. 2, pp. 173–195, 2002.
[23] I. G. Damousis, A. Bakirtzis, and P. Dokopoulos, "A solution to the unit commitment problem using integer-coded genetic algorithm," *IEEE Transactions on Power Systems*, vol. 19, pp. 1165–1172, 2004.
[24] A. Viana, J. Sousa, and M. Matos, "Fast solutions for UC problems by a new metaheuristic approach," *Electric Power Systems Research*, vol. 78, pp. 1385–1395, 20087.
[25] Y.-W. Jeong, J.-B. Park, S.-H. Jang, and K. Lee, "A new quantum-inspired binary pso: Application to unit commitment problems for power systems," *Power Systems, IEEE Transactions on*, vol. 25, no. 3, pp. 1486 –1495, aug. 2010.
[26] T. Lau, C. Chung, K. Wong, T. Chung, and S. Ho, "Quantum-inspired evolutionary algorithm approach for unit commitment," *Power Systems, IEEE Transactions on*, vol. 24, no. 3, pp. 1503 –1512, aug. 2009.
[27] C. Chung, H. Yu, and K. P. Wong, "An advanced quantum-inspired evolutionary algorithm for unit commitment," *Power Systems, IEEE Transactions on*, vol. 26, no. 2, pp. 847 –854, may 2011.
[28] J. Ebrahimi, S. Hosseinian, and G. Gharehpetian, "Unit commitment problem solution using shuffled frog leaping algorithm," *Power Systems, IEEE Transactions on*, vol. 26, no. 2, pp. 573 –581, may 2011.
[29] M. Moghimi Hadji and B. Vahidi, "A solution to the unit commitment problem using imperialistic competition algorithm," *Power Systems, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2011.

Ana Viana (El.& Comp. Eng., Ph.D., 2004) is currently Full Professor at the School of Engineering, Polytechnic Institute of Porto, Portugal, and Senior Researcher at INESC Porto.

João Pedro Pedroso (Eng. Math., Ph.D., 1996) is currently Assistant Professor at the Faculty of Sciences, University of Porto, Portugal, and Senior Researcher at INESC Porto.