MILP-Based Algorithm for the Global Solution of Dynamic Economic Dispatch Problems with Valve-Point Effects

Loïc Van Hoorebeeck
ICTEAM
UCLouvain
Louvain-la-Neuve, Belgium
Email: loic.vanhoorebeeck@uclouvain.be

P.-A. Absil
ICTEAM
UCLouvain
Louvain-la-Neuve, Belgium

Anthony Papavasiliou
CORE
UCLouvain
Louvain-la-Neuve, Belgium

***Abstract*—The Dynamic Economic Dispatch (DED) problem consists in satisfying a certain demand for electric power among scheduled generating units over a certain interval of time while satisfying the operating constraints of these units. The consideration of the valve-point effect (VPE) makes the problem more practical but also more challenging due to the non-linear and non-smooth constraints that are required for representing the model. We present a method, based on a sequence of piecewise linear approximations, which produces a feasible solution along with a lower bound on the global solution. In this way, this deterministic approach can trade off the speed which characterizes certain heuristics that are usually used to solve the DED-VPE for a better solution and insights about the problem. The method is applied to a widely used case study and provides a lower solution objective than the best known solution to date.**

I. Introduction

The Economic Dispatch (ED) problem is an important optimization problem in short-term power system planning. It consists in the optimal dispatch of power among scheduled electricity generation facilities in order to meet the system load at a minimal cost. Commonly, the fuel cost functions have been modeled as a smooth quadratic function in the ED problem. Unfortunately, such a model does not reflect the valve-point effect (VPE), i.e., the fact that turbines operating off a valve point run less efficiently due to throttling losses. This significantly affects the output of facilities that are now characterized by a non-smooth and non-convex cost function. The latter characteristics of the cost function prevent us from using traditional derivative-based optimization techniques for solving the problem.

In the past decades, a plethora of methods have been developed in order to address this problem, including neural networks [1], simulated annealing (SA) [2], genetic algorithms (GA) [3], evolutionary progamming [4], differential evolution (DE) [5] and particle swarm optimization (PSO) [6]. A more exhaustive list of these methods and other hybrid combinations can be found in [7]. Most of the aforementioned techniques are heuristics and, if they often give a fast and reasonable solution, they lack guarantees with respect to the returned solution.

On the other side, deterministic mathematical programming-based optimization techniques have the advantage of providing information about the distance of the solution from optimality. Such methods have been developed by piecewise linearization of the VPE-term [8] and of the entire objective function [9], leading to respectively a mixed integer quadratic programming (MIQP) and a mixed integer linear programming (MILP) problem. More recently in [10], an adaptive MIQP method has been proposed with the significant feature of providing a *global* solution of the static ED. This method relies on a sequence of under-approximations for which the sequence of optimal solutions eventually converges to the global solution of the original problem. Early developments of the adaptive approach can be found in the technical report [11].

Here, we follow this adaptive approach. Our contributions are to (i) apply this approach on the dynamic ED and (ii) to investigate the benefit of using a MILP formulation.

The remainder of this paper is organized as follows. In Section II, the full problem is introduced and the VPE is described. The linear version of the adaptive method is presented in Section III. Then the application on a 10-generator case study over 24 hours is performed in Section IV and the cost of neglecting the VPE is computed. Finally, conclusions are drawn in Section V.

II. Problem Statement

This section outlines the problem of interest, namely the dynamic economic dispatch (DED) problem. This problem consists in minimizing the fuel costs of the thermal power units throughout a certain time period subject to operational constraints. The objective is defined as the sum of the individual cost functions over each time step and is therefore separable,

$$f(\mathbf{p}) = \sum_{i=1,\, t=1}^{n,\, T} f_{it}(p_{it})\,, \tag{1}$$

where f is the total cost function ($\$/\mathrm{h}$), f_{it} the fuel cost associated with generator i at time t and $\mathbf{p}$ is the stacked production vector of each individual generator production p_{it}

(MW). A common model of the fuel cost functions including the VPE is the sum of a smooth quadratic part and a non-smooth rectified sine, i.e.,

$$f_{it}(p_{it}) = a_i p_{it}^2 + b_i p_{it} + c_i + d_i \left|\sin e_i(p_{it} - p_i^{\min})\right| , \quad (2)$$

with appropriate parameters a_i, b_i, c_i, d_i, e_i. The impact of the VPE, namely the non-smooth and high multimodal nature of the problem, is underlined in Figure 2. The model must of course enforce power balance which couples the optimization problem,

$$\sum_{i=1}^{n} p_{it} = P_t^D + p^L(\mathbf{p}_t)\,, \quad (3)$$

with P_t^D and $p^L(\mathbf{p}_t)$ respectively the load demand at time t and the transmission lost in MW. The latter is not included in the method presented here and the assumption $p^L(\mathbf{p}_t) = 0$ is made throughout the rest of this paper. This extension is left for future research.

We consider spinning upward reserve, i.e., extra generating capacity in case of contingencies,

$$\sum_{i=1}^{n} s_{it} \geq S_t\,, \quad (4)$$

$$s_{it} \leq R_i^U\,, \quad (5)$$

with s_{it} the extra capacity that generator i must be able to provide and S_t the total spinning upward reserve required at time t.

Finally, the optimization problem is subject to operational constraints such as the admissible range of power production,

$$P_i^{\min} \leq p_{it}\,, \quad (6)$$

$$p_{it} + s_{it} \leq P_i^{\max}\,, \quad (7)$$

and ramp constraints,

$$-R_i^D \leq p_{it} - p_{i(t-1)} \leq R_i^U\,, \quad (8)$$

with $P_i^{\min}$ and $P_i^{\max}$ the minimum and maximum acceptable range of power production and R_i^D, R_i^U the downward and upward ramp rate limit of the i-th generator.

Note that, with the exception of power loss, the method described in the following section can be easily extended to more complicated models that account for downward spinning reserve, multiple fuels and prohibited operating zones.

III. Method description

This section is devoted to the characterization of an algorithm for the solution of the DED. The method consists of a sequence of piecewise linear approximations, the surrogate problems, tackled by a MILP solver and is described in Figure 1. Let us now define the surrogate problem.

A. Surrogate problem

For all i and t, let $\mathbf{X}_{it}$ be a set of points $X_{it1} < X_{it2} < \cdots < X_{itn_{it}^{\text{knot}}}$, called knots, from which we construct a piecewise linear approximation g_{it} of f_{it}. We create the original set of knots of unit i equally for each time t as the union of two subsets. The first subset is the set of kink points, which are the points where the cost function is non-smooth. The second subset is the set of local maxima of the rectified sine. Hence for every unit i and time t, the set of initial knots is equal to

$$X_{itj} = P_i^{\min} + \frac{(j-1)\pi}{2e_i} \qquad j = 1 \ldots n_{it}^{\text{knot}}\,, \quad (9)$$

with $n_{it}^{\text{knot}} = 1 + \lceil (P_i^{\max} - P_i^{\min})\frac{2e_i}{\pi}\rceil$. We then construct the surrogate approximation g_{it} through a binary formulation, i.e.,

$$g_{it}(p_{it}) := \begin{cases} \sum_{j=1}^{n_{it}^{\text{knot}}-1} \alpha_{itj}\xi_{itj} + \eta_{itj}\beta_{itj}\,, \\ \text{with } \sum_{j=1}^{n_{it}^{\text{knot}}-1} \xi_{itj} = p_{it}\,, \\ \quad \sum_{j=1}^{n_{it}^{\text{knot}}-1} \eta_{itj} = 1\,, \quad \eta_{itj} \in \{0,1\}\,, \\ \quad X_{itj}\,\eta_{itj} \leq \xi_{itj} \leq X_{itj+1}\,\eta_{itj}\,, \end{cases} \quad (10)$$

where α_{itj} and β_{itj} are the slope and vertical intercept of the linear pieces; see [10] for details of the concept. The binary variables η act as switches which select the different pieces. Following (1), we also define $g(\boldsymbol{p}) := \sum_{i=1,\,t=1}^{n,\,T} g_{it}(p_{it})$. The surrogate optimization problem becomes

$$\begin{aligned} \min_{\eta_{itj},\,\xi_{itj},\,p_{it},\,s_{it}} \; & \sum_{i=1,\,t=1}^{n,\,T}\sum_{j=1}^{n_{it}^{\text{knot}}-1} \alpha_{itj}\xi_{itj} + \beta_{itj}, \\ \text{subject to } & \sum_{i=1}^{n} p_{it} = P_t^D, \\ & \sum_{i=1}^{n} s_{it} \geq S_t, \\ & s_{it} \leq R_i^U, \\ & P_i^{\min} \leq p_{it}, \\ & p_{it} + s_{it} \leq P_i^{\max}, \\ & -R_i^D \leq p_{it} - p_{i(t-1)} \leq R_i^U, \\ & \sum_{j=1}^{n_{it}^{\text{knot}}-1} \xi_{itj} = p_{it}, \\ & \sum_{j=1}^{n_{it}^{\text{knot}}-1} \eta_{itj} = 1, \quad \eta_{itj} \in \{0,1\}, \\ & X_{itj}\eta_{itj} \leq \xi_{itj} \leq X_{itj+1}\eta_{itj}. \end{aligned} \quad (11)$$

B. Knot Update Mechanism and Algorithm Statement

Assume a solution of the surrogate problem has been obtained. If the gap between the true and surrogate objective function evaluated at this point is too large, an increase in the number of knots should be contemplated. Indeed, the approximation will be enhanced and the same is true for the

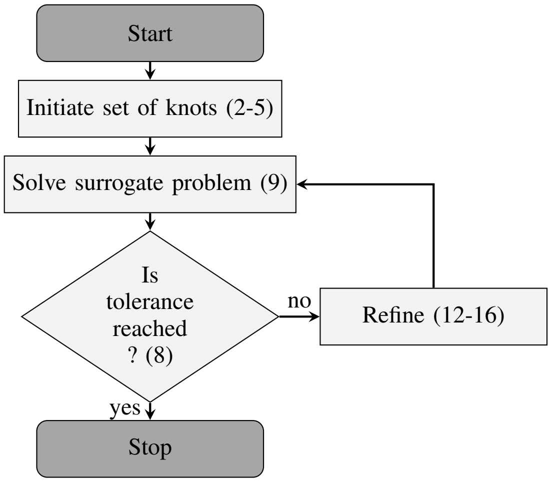

Fig. 1. Flow chart of the adaptive piecewise-linear approximation (lines reference to Algorithm 1).

surrogate solution. In [9], the adopted approach was to increase the knot sampling over the entire allowable range. However, this increases the number of knots and therefore the number of binary variables exponentially. In this work, we follow the knot update mechanism from [10], i.e., the previous surrogate solution is added to the knot list. As a consequence, in the new iteration, the surrogate solution differs from the old one and convergence is guaranteed. The proposed APLA algorithm (Algorithm 1) is very similar to Algorithm 1 in [10].

Algorithm 1 APLA: Adaptive piecewise-linear approximation

1: Set tolerance parameter δ_{tol}
2: **for** $i := 1 \dots n$ **do**
3:     **for** $t := 1 \dots T$ **do**
4:         Choose set of knots ($\mathbf{X}_{it}$) including kink points
5:         $\mathbf{G}_{it} \leftarrow f_{it}(\mathbf{X}_{it})$
6:     **end for**
7: **end for**
8: **while** $\tilde{\delta}^k > \delta_{\text{tol}}$ **do**
9:     $\mathbf{p^k} \leftarrow$ optimal solution of MILP surrogate problem (11), obtained with MILP solver with tolerance γ
10:     $\delta^k \leftarrow f(\mathbf{p^k}) - \mathbf{g^k}(\mathbf{p^k})$
11:     $\tilde{\delta}^k \leftarrow \min_{l=1\dots k} f(\mathbf{p^l}) - \mathbf{g^k}(\mathbf{p^k})$
12:     **for** $i := 1 \dots n,\, t := 1 \dots T$ **do**
13:         **if** $\min_{j \in \{1 \dots n_{it}^{\text{knot}}\}} \left| p_{it}^k - X_{itj} \right| > 0$ **then**
14:             $\mathbf{X}_{it} \leftarrow \text{insort}(\mathbf{X}_{it},\, p_{it}^k)$ ▷ ordered insertion
15:             $\mathbf{G}_{it} \leftarrow \text{insert}(\mathbf{G}_{it},\, f_{it}(p_{it}^k))$ ▷ insert at same index as previous line
16:             $n_{it}^{\text{knot}} \leftarrow n_{it}^{\text{knot}} + 1$
17:         **end if**
18:     **end for**
19: **end while**
20: **return** $\arg\min_{l=1\dots k} f(\mathbf{p^l})$

C. Bound to optimal solution

At step k of Algorithm 1, the objective function evaluated at the optimal solution $\boldsymbol{p}^*$ can be bounded as follows,

$$\min_{l=1\dots k} f(\boldsymbol{p}^l) - \tilde{\delta}^k - \gamma^k - \epsilon^k \leq f(\boldsymbol{p}^*) \leq \min_{l=1\dots k} f(\boldsymbol{p}^l)\,, \quad (12)$$

where $\tilde{\delta}^k$ is the gap between the best known objective and the surrogate function at point $\mathbf{p^k}$ as computed in line 11 of Algorithm 1, γ^k is the tolerance set to the solver and ϵ^k represents the over-approximation error. Let us explain more precisely each of these bounds and how it evolves as k increases.

a) The over-approximation error (ϵ^k) : Algorithm 1 is based on a sequence of piecewise linear approximations. In contrast with the APQUA method of [10], the approximation is not guaranteed to be an under-approximation. It can be seen that the approximation will be an under-approximation if the true cost function is concave in each segment delimited by the initial knots. However, this concavity assumption is not satisfied in our case because the curvature of the rectified sine vanishes at the kink points (see bottom right magnification in Figure 2). More precisely, function f_{it} is convex on $X_{itj} + \frac{1}{e_i} \arcsin\left(\frac{2a_i}{d_i e_i^2}\right) \times [-1, 1]$. In the example studied in Section IV, these convex parts are very small: about 0.5% of the whole domain. The error ϵ^k can be computed as

$$\epsilon^k = \max_{\mathbf{p}} (g^k(\boldsymbol{p}) - f(\boldsymbol{p})), \quad (13)$$

$$= \sum_{i=1,\, t=1}^{n,\, T} \underbrace{\max_{P_i^{\min} \leq p_{it} \leq P_i^{\max}} (g^k(p_{it}) - f(p_{it}))}_{:= \epsilon_{it}^k}\,. \quad (14)$$

Note that this last equation can be easily calculated at every iteration since it can be viewed as $n \times T \times n^{\text{knot}}$ decoupled optimization problems of a single variable. Besides, as we never remove points, $(\epsilon^k)_{k \in \mathrm{N}}$ can be bounded above; we obtain for all $k = 1,\, 2,\, \dots$

$$\epsilon^k \leq \epsilon^{\max} := \sup_{g \in \mathcal{G}} \max_{\mathbf{p}} (g(\mathbf{p}) - f(\mathbf{p}))\,, \quad (15)$$

where $\mathcal{G}$ is the set of all piecewise interpolations of f such that the kink points belong to the set of knots that defines the interpolation. In other words, $\mathcal{G}$ is the set of functions to which Algorithm 1 has access in order to approximate f. For the case study investigated in Section IV, we get $\epsilon^{\max} = 0.32\,\$$ which is very small with respect to the other bounds.

b) The gap between the best objective and the surrogate function ($\tilde{\delta}^k$) : Following [10], we show in Theorem 1 that $(\delta^k)_{k \in \mathrm{N}}$, the gap between the objective and surrogate function at point $\boldsymbol{p}^k$, converges to 0. Then, we use this result to prove that the limit superior of $(\tilde{\delta}^k)_{k \in \mathrm{N}}$ goes to zero as well.

Theorem 1. $\lim_{k \to \infty} \delta^k = 0$

Proof. We first show that f and g^k, $k = 0,\, 1,\, \dots$ are Lipschitz continuous on the feasible set.

For each unit i and time t, we have $|f_{it}(p + \Delta) - f_{it}(p)| \leq (2a_i P_i^{\max} + b_i + d_i e_i)\Delta := K_i\,\Delta$ with K_i the so-called Lipschitz constant. Summing up on each unit, $K := T \times \sum_{i=1}^n K_i$ is a valid Lipschitz constant for f. Since g_{it}^k is a continuous piecewise interpolation of f_{it}, it is also Lipschitz continuous and K_i (resp. K) is a valid Lipschitz constant for g_{it}^k (resp.

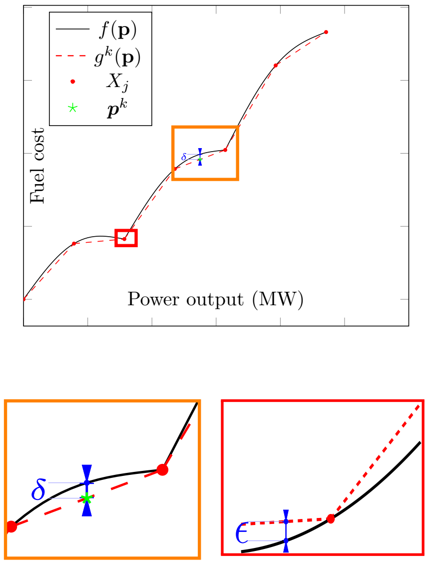

Fig. 2. Outline of the true f and surrogate g^k functions. The bottom left magnification allows a better vizualization while the bottom right one shows a tiny convex zone around a kink point.

g^k). Let $(\mathbf{p}^k)_{k\in\mathrm{N}}$ be the sequence of optimal solutions of the surrogate problem associated with function g^k, we then obtain

$$\begin{aligned}\delta^k &= f(\mathbf{p}^k) - g^k(\mathbf{p}^k)\,,\\ &= f(\mathbf{p}^k) - g^k(\mathbf{p}^{k-1}) + g^k(\mathbf{p}^{k-1}) - g^k(\mathbf{p}^k)\,,\\ &= f(\mathbf{p}^k) - f(\mathbf{p}^{k-1}) + g^k(\mathbf{p}^{k-1}) - g^k(\mathbf{p}^k)\,,\\ &\leq 2K\,\left|\left|\mathbf{p}^k - \mathbf{p}^{k-1}\right|\right|\,,\end{aligned}$$

where the 3rd line comes from the knot updating criterion and the last line from the Lipschitz continuity.

Suppose for contradiction that $(\delta^k)_{k\in\mathrm{N}}$ does not converge to 0. Then there is $\delta^* > 0$ and an infinite subsequence $(\delta^{k_j})_{j\in\mathrm{N}}$ such that $|\delta^{k_j}| > \delta^*$ for all j. Then, given any j, we have that for all $J > j$, $\|\boldsymbol{p}^{m_J} - \boldsymbol{p}^{m_j}\| \geq \delta^*/(2K)$. This implies that the subsequence $(p^{m_j})_{j\in\mathrm{N}}$ is unbounded, a contradiction with the admissible range constraints. □

The bottom left magnification in Figure 2 depicts an example of δ^k, note that the i and t indices have been omitted. Finally, due to the definition of $\tilde{\delta}^k$, it immediately follows from Theorem 1 that

$$\limsup_{k\to\infty} \tilde{\delta}^k \leq 0\,. \tag{16}$$

c) The solver tolerance (γ^k) : The sequence $\left(\gamma^k\right)_{k\in\mathrm{N}}$ is not monotonic and is bounded below by $\gamma f(\mathbf{p}^*)$ with γ the solver relative tolerance gap to the global solution of the surrogate problem.

D. Extension to broader class of functions

The method, as written in Algorithm 1, works for piecewise smooth-concave functions f_{it}. Nevertheless, it can be extended to convex functions. In a similar fashion as the outer approximation (OA) algorithm [12], the under approximation of the convex part is tackled by adding constraints instead of variables. Let us apply this procedure on a simple example, in order to illustrate the approach: we consider the problem

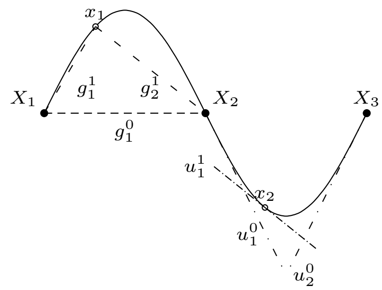

Fig. 3. APLA extension to non-concave function.

$$\min_{x\in[0,2\pi]} \sin(x)\,. \tag{17}$$

Starting from the three knots $(X_1, X_2, X_3) = (0, \pi, 2\pi)$, the surrogate problem is written as follows

$$\begin{aligned}\min_{\boldsymbol{\xi},\boldsymbol{\eta},t}\quad & g_1^0(\xi_1,\eta_1) + t\,,\\ \text{s.t.}\quad & X_0\eta_1 \leq \xi_1 \leq X_1\eta_1\,,\\ & X_1\eta_2 \leq \xi_2 \leq X_2\eta_2\,,\\ & u_1^0(\xi_2,\eta_2) \leq t\,,\\ & u_2^0(\xi_2,\eta_2) \leq t\,,\\ & \eta_1 + \eta_2 = 0\,,\quad \eta_{1,2} \in \{0,1\}\ ,\end{aligned}$$

where $g_j^k(\xi,\eta) = \alpha\xi + \beta\eta$ and (α, β) defines the line g_j^k from Figure 3. The same idea applies for u_j^k.

Let us analyse the procedure of refining around a point both in the concave (x_1) and in the convex region (x_2). For the first case, we simply add x_1 to the set of knots, splitting g_1^0 into g_1^1 and g_2^1. And, for the second case, we request t to be also above the tangent in x_2, i.e. $t \geq u_1^1(\xi_2, \eta_2)$. Doing so we can, with very few changes to Algorithm 1, adapt the method to deal with any piecewise-smooth function.

IV. Test Case Study

In this section, a 10-unit DED without losses over $T = 24$ hours is studied and the obtained solution is compared with the solution that is obtained by ignoring the VPE. The data set used for the case study can be found in [13] and the spinning reserve is set at 5 % of the demand. The optimization is performed on a computer with an Intel-i7 CPU and 16 GB of RAM. Gurobi 8.0.0 has been used with a relative gap tolerance of $\gamma = 0.25\%$ and the model has been coded in AMPL. Note that a feasible solution stays feasible for the surrogate problem at every iteration. Hence, in order to benefit from the previous iterations, the MILP solver is fed with the best known solution of the true problem as an initial incumbent. Algorithm 1 is also slightly improved by asking, in line 9, the solver to return the 50 best incumbent solutions instead of the sole best one. Then, the best candidate solution becomes the minimum with respect to the true objective value of this set of incumbent solutions, provided that it is smaller than the actual best candidate.

After 9 iterations and 902 seconds, a solution with $\tilde{\delta} = 1.42$ \$ is found with objective 1016276 \$. This is an improvement over the previous best solution in the literature with objective

1016311 $ [7]. Neglecting $\epsilon^{\max}$, the final relative optimality gap is

$$\frac{|f(\mathbf{p}^*) - \mathbf{f}(\mathbf{p})|}{f(\mathbf{p})} = \gamma + \frac{\delta}{f(\mathbf{p})} = 0.25\%. \quad (18)$$

Following standard practice of the VPE literature [6], [7], [10], we include the power dispatch among the generating units in Table I in order to validate our results.

The practical economy of the consideration of the VPE can now be computed. If the parameters d and e from Eq. 2 are set to 0, i.e. if the VPE is ignored, we face a convex quadratic programming (QP) problem. When the solution of the QP problem is inserted into the real objective function, we obtain an objective of 1036211\$. Hence, the additional work for taking into account the VPE decreases the cost by 1.96%.

V. Conclusion

In this paper, we have presented a deterministic method which accounts for valve-point effect in the dynamic economic dispatch problem. The method relies on a succession of piecewise linear approximations of the cost function which define a surrogate problem. The method is adaptive in the sense that the previously computed best candidate minimum of the surrogate problem is added to the knots used to define the approximation. Doing so, the method converges to a solution which is optimal within the mixed-integer solver accuracy, plus an ϵ term that can be bounded a priori and turns out to be tiny in problem instances found in the literature.

The result of our case study confirms this distinctive feature by slightly improving the best known solution to the considered problem. However, as each surrogate problem remains challenging, the computation time of our approach is higher – approximately 10 times – than the method giving the previously best results. The proposed method benefits from an another advantage: it is not limited to the dynamic economic dispatch problem, as any optimization problem with separable objective can be treated in a similar way.

Future work may involve an acceleration of the method, for instance by using the previous solutions in order to restrict the dispatch range of the units to a neighborhood of the optimal solution and thereby reduce the number of variables that are used in the formulation. Moreover, the method can be extended to a unit commitment formulation of the problem, which expands its scope to day-ahead scheduling applications where we are also afforded a larger run time.

Acknowledgment

This work was supported by the Fonds de la Recherche Scientifique - FNRS under Grant no. PDR T.0025.18.

References

[1] T. Yalcinoz, B. Cory, and M. Short, "Hopfield neural network approaches to economic dispatch problems," *International Journal of Electrical Power & Energy Systems*, vol. 23, no. 6, pp. 435 – 442, 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0142061500000843

[2] K. P. Wong and C. C. Fung, "Simulated annealing based economic dispatch algorithm," *IEEE Proceedings C - Generation, Transmission and Distribution*, vol. 140, no. 6, pp. 509–515, Nov 1993.

[3] W. N. W. Abdullah, H. Saibon, A. A. M. Zain, and K. L. Lo, "Genetic algorithm for optimal reactive power dispatch," in *Proceedings of EMPD '98. 1998 International Conference on Energy Management and Power Delivery (Cat. No.98EX137)*, vol. 1, March 1998, pp. 160–164 vol.1.

[4] K. S. Swamp and A. Natarajan, "Constrained optimization using evolutionary programming for dynamic economic dispatch," in *Proceedings of 2005 International Conference on Intelligent Sensing and Information Processing, 2005.*, Jan 2005, pp. 314–319.

[5] R. E. Perez-Guerrero and J. R. Cedeno-Maldonado, "Economic power dispatch with non-smooth cost functions using differential evolution," in *Proceedings of the 37th Annual North American Power Symposium, 2005.*, Oct 2005, pp. 183–190.

[6] C. H. Chen and S. N. Yeh, "Particle swarm optimization for economic power dispatch with valve-point effects," in *2006 IEEE/PES Transmission Distribution Conference and Exposition: Latin America*, Aug 2006, pp. 1–5.

[7] S. Pan, J. Jian, and L. Yang, "A hybrid MILP and IPM approach for dynamic economic dispatch with valve-point effects," *International Journal of Electrical Power & Energy Systems*, vol. 97, pp. 290 – 298, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0142061517305744

[8] M. Q. Wang, H. B. Gooi, S. X. Chen, and S. Lu, "A mixed integer quadratic programming for dynamic economic dispatch with valve point effect," *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2097–2106, Sept 2014.

[9] S. Pan, J. Jian, and L. Yang, "A mixed integer linear programming method for dynamic economic dispatch with valve point effect," 2017, arXiv:1702.04937.

[10] P. A. Absil, B. Sluysmans, and N. Stevens, "MIQP-based algorithm for the global solution of economic dispatch problems with valve-point effects," in *2018 Power Systems Computation Conference (PSCC)*, June 2018, pp. 1–7.

[11] M. Azzam, S. E. Selvan, A. Lefèvre, and P.-A. Absil, "Mixed integer programming to globally minimize the economic load dispatch problem with valve-point effect," http://sites.uclouvain.be/absil/2014.08, July 2014.

[12] M. A. Duran and I. E. Grossmann, "An outer-approximation algorithm for a class of mixed-integer nonlinear programs," *Math. Program.*, vol. 36, no. 3, pp. 307–339, Oct. 1986. [Online]. Available: https://doi.org/10.1007/BF02592064

[13] P. Attaviriyanupap, H. Kita, E. Tanaka, and J. Hasegawa, "A hybrid EP and SQP for dynamic economic dispatch with nonsmooth fuel cost function," *IEEE Transactions on Power Systems*, vol. 17, no. 2, pp. 411–416, May 2002.

TABLE I
Objective value: 1016276 $.

Hour	U1 (MW)	U2	U3	U4	U5	U6	U7	U8	U9
1	150	143.56	185.533	60	122.867	122.45	129.59	47	20
2	150	223.56	229.399	60	73	122.45	129.59	47	20
3	150	303.56	297.399	60	73	122.45	129.59	47	20
4	226.624	316.799	305.67	60	122.867	122.45	129.59	47	20
5	226.624	396.799	299.67	60	122.867	122.45	129.59	47	20
6	303.248	396.799	321.179	60	172.733	122.45	129.59	47	20
7	303.248	396.799	297.399	107.913	222.6	122.45	129.59	47	20
8	379.873	396.799	297.399	149.87	172.733	122.45	129.59	52.285	20
9	456.497	396.799	297.399	191.246	172.733	122.45	129.59	82.285	20
10	456.497	396.799	303.399	241.246	222.6	160	129.59	85.312	21.556
11	456.497	396.799	297.399	291.246	222.6	160	129.59	85.312	51.556
12	456.497	460	307.698	291.246	222.6	160	129.59	85.312	52.057
13	456.497	396.799	302.899	241.246	222.6	160	129.59	85.312	22.057
14	456.497	396.799	294.373	191.246	172.733	122.45	129.59	85.312	20
15	379.873	396.799	303.397	168.624	122.867	122.85	129.59	77	20
16	303.248	393.821	291.266	118.624	73	122.45	129.59	47	20
17	303.248	313.821	297.399	68.624	122.867	122.45	129.59	47	20
18	379.873	393.821	297.399	60	122.867	122.45	129.59	47	20
19	456.497	396.799	297.399	70.219	172.733	122.45	129.59	55.312	20
20	456.497	460	332.782	120.219	222.6	160	129.59	85.312	50
21	456.497	389.533	297.399	110	222.6	158.069	129.59	85.312	20
22	379.873	309.533	297.399	60	172.733	122.45	129.59	81.422	20
23	303.248	229.533	237.89	60	122.867	122.45	129.59	51.422	20
24	226.624	222.266	178.22	60	122.867	122.45	129.59	47	20