A Dual Bisection Approach to Economic Dispatch of Generators With Prohibited Operating Zones

Lucrezia Manieri, *Member, IEEE*, Alessandro Falsone, *Member, IEEE*, and Maria Prandini, *Fellow, IEEE*

***Abstract*—We address economic dispatch of power generators with prohibited operating zones. The problem can be formulated as an optimization program with a quadratic cost, non-convex local operating constraints, and a scalar quadratic coupling constraint accounting for load demand and power losses. A duality-based resolution approach integrating a bisection iterative scheme is proposed to reduce computational complexity while guaranteeing finite time feasibility of the primal iterates and a cost improvement throughout iterations. Extensive simulations show that the approach outperforms state-of-the-art competitors and consistently computes feasible primal solutions with a close-to-zero optimality gap at a low computational cost.**

***Index Terms*—Energy systems, large-scale systems, optimization algorithms.**

I. Introduction

UNIT Commitment (UC) and Economic Dispatch (ED) are crucial for power systems operation. UC [1] determines the generating units that will be possibly activated to produce the forecasted electricity demand along some reference time horizon, while ED [2] allocates the demand in each time slot by defining the actual amount of power that each of the committed generators has to produce in that time slot. The ED problem can be solved after the UC problem or jointly, in an integrated manner.

The ED problem admits several formulations (see [3, Ch. 7] and [4] for an overview), that can differ for the objective function (e.g., reduction of fuel costs and emissions of greenhouse gases), the adoption of multiple objectives, the characteristics of the generators (e.g., multiple fuel options, presence of prohibited operating regions), etc. Many of these formulations involve discrete decision variables, which make the resulting optimization problem combinatorial, with a complexity that grows exponentially in the number of generators.

Manuscript received 8 March 2024; revised 7 May 2024; accepted 24 May 2024. Date of publication 3 June 2024; date of current version 28 June 2024. Recommended by Senior Editor K. Savla. *(Corresponding author: Maria Prandini.)*

The authors are with the Dipartimento di Elettronica Informazione e Bioingegneria, Politecnico di Milano, 20133 Milano, Italy (e-mail: lucrezia.manieri@polimi.it; alessandro.falsone@polimi.it; maria.prandini@polimi.it).

Digital Object Identifier 10.1109/LCSYS.2024.3409230

State-of-the-art methods for ED resort to a wide gamut of paradigms, ranging from stochastic (possibly hybrid) heuristic methods [5], [6], [7], to exact resolution schemes based on implicit enumeration [8], and linearization of quadratic and nonlinear terms [9]. Although their performance depends on the chosen problem formulation, they typically either lack convergence guarantees or do not scale with the size of the problem, thus becoming prohibitive in practice.

In this letter, we consider an ED problem where we aim at minimizing the cost to supply the required energy demand via generators with prohibited operating regions. Following [6], we formulate the problem as an optimization program with continuous and binary decision variables, quadratic cost, non-convex local constraints, and a scalar quadratic coupling constraint taking into account power losses. The intrinsic combinatorial nature of the problem that is hidden in the non-convex constraints makes it hard to solve and calls for suitable resolution schemes. Here, we propose a duality-based approach integrating the bisection iterative scheme in [10] to tackle computational complexity, while guaranteeing finite time feasibility of the primal iterates and a cost that is not increasing throughout iterations.

Resolution strategies for similar ED programs have been proposed in the literature using linear approximation, direct search and stochastic optimization methods. In [9], [11] the nonlinear terms in the constraints (and, possibly, the cost) are linearized or approximated via piece-wise affine functions to obtain a mixed-integer linear formulation of the problem. Such reformulation, however, typically includes several additional binary variables, that ultimately increase the combinatorial complexity of the problem. In [5] and [12], the optimal solution is searched in the (non-convex) feasibility set of the problem via a Genetic Algorithm (GA), using different selection and recombination strategies. Parallel streams of work investigated the use of spatial Particle Swarm Optimization (PSO) and Evolutionary Programming (EP) methods, see [6], [13], and [14]. A distributed approach combining flooding-based consensus and a differential evolution algorithm was proposed in [15]. All these strategies use a stochastic approach to tackle the non-convexity of the optimization program and are not too complex to implement. However, they typically lack finite-time convergence guarantees to a feasible solution and their performance highly depends on their initialization, thus requiring multiple runs to find good-quality solutions. A deterministic two-level Branch-and-Bound method is, instead, proposed in [8] to gain efficiency by exploiting the problem structure. However, the resulting method is only partly effective since it shares all the critical aspects of standard Branch-and-Bound approaches and does not scale with the size of the problem.

Our approach exploits duality to gain in computational efficiency. Note that duality is also used in [16] but to solve a multi-objective ED problem without considering generators with prohibited operating regions. A weighted sum method is used to recast the multi-objective cost as a single-objective cost, using a bisection-based heuristic to determine the value of the weights in the sum for which the Pareto front is smooth and uniform. Duality allows to take advantage of the structure of the resulting problem and formulate it as a quadratic program, which is easy to solve.

The remainder of this letter is structured as follows. Section II provides the formulation for the ED problem based on [6], whilst the proposed approach is introduced in Section III and tested via numerical simulations in Section IV. Finally, Section V concludes this letter.

II. Problem Formulation

Consider an Economic Dispatch Problem (EDP) with m generator units, each producing a power $P_i \geq 0$, which is zero if the generator i does not participate in the energy provision at the considered time slot and is positive otherwise. The problem is modeled according to the widely-adopted mixed-integer quadratic formulation proposed in [6]. Generators are assumed to use a single type of fuel and the cost J_i for unit i to produce an amount of power P_i is given by the quadratic cost function

$$J_i(P_i) = \omega_{i0} + \omega_{i1} P_i + \omega_{i2} P_i^2, \tag{1}$$

where ω_{i0}, ω_{i1} and ω_{i2} are positive scalar coefficients specific for generator i, $i = 1, \dots, m$.

Generators can have *prohibited zones* within their domain of operation, due to physical limitations of individual power plant components [17]. This is the case, for example, for the vibrations in a shaft bearing that are amplified at certain operating regimes [5], which should then be avoided. This is modeled by associating to each generator i a set of operating regions $\{[\underline{P}_{i,j}, \bar{P}_{i,j}], j = 1, \dots, N_i\}$ such that $\underline{P}_{i,1} = P_i^{\min} > 0$, $\bar{P}_{i,N_i} = P_i^{\max} > P_i^{\min}$, $\underline{P}_{i,j} < \bar{P}_{i,j}$, and $\bar{P}_{i,j-1} < \underline{P}_{i,j}$.

The power P_i produced by generator i will then satisfy (only) one of the following conditions

$$P_i = 0, \tag{2a}$$

$$\underline{P}_{i,j} \leq P_i \leq \bar{P}_{i,j} \quad \text{for some } j \in \{1, \dots, N_i\}. \tag{2b}$$

The operation of a generator may also be subject to ramp limits with the purpose of avoiding abrupt variations in the power output between the previous time slot and the current one. If we denote as $P_{i,0}$ the power provided by generator i in the previous time slot, ramp limits can be enforced requiring

$$-\underline{\Delta}_i \leq P_i - P_{i,0} \leq \bar{\Delta}_i, \tag{3}$$

with $\bar{\Delta}_i > 0$ and $\underline{\Delta}_i > 0$ denoting the maximum power increase and decrease, respectively, that generator i can allow.

In order to satisfy a certain power demand $P_d \geq 0$, the power of all generators should be set so as to satisfy the following (scalar) constraint

$$\sum_{i=1}^{m} P_i - P_\ell(\vec{P}) \geq P_d, \tag{4}$$

that accounts also for the power losses $P_\ell(\vec{P})$, with $\vec{P} = [P_1 \cdots P_m]^\top$, which can be computed using Kron's loss formula [3, Sec. 7.6] as

$$\mathcal{P}_\ell(\vec{P}) = \sum_{i=1}^{m} \sum_{j=1}^{m} b_{ij} P_i P_j + \sum_{i=1}^{m} b_{i0} P_i + b_{00}, \tag{5}$$

with b_{ij}, b_{i0}, b_{00}, $i, j = 1, \dots, m$, being suitable coefficients.

Let $\mathcal{P}_i$ be the set

$$\mathcal{P}_i = \left[P_{i,0} - \underline{\Delta}_i, P_{i,0} + \bar{\Delta}_i\right] \cap \left(\{0\} \cup_{j=1}^{N_i} \left[\underline{P}_{i,j}, \bar{P}_{i,j}\right]\right) \tag{6}$$

defining the feasible power output values for generator i and denote with $\tilde{P}_i^{\max}$ its maximum admissible power output, i.e., $\tilde{P}_i^{\max} = \max_{P_i \in \mathcal{P}_i} P_i \leq P_i^{\max}$. Clearly, for (4) to be admissible, the power demand P_d shall be less than the maximum power that can be produced by all generators minus the corresponding losses, i.e., $P_d < \sum_{i=1}^{m} \tilde{P}_i^{\max} - P_\ell(\vec{\tilde{P}}^{\max})$, with $\vec{\tilde{P}}^{\max} = [\tilde{P}_1^{\max} \cdots \tilde{P}_m^{\max}]^\top$.

Then, the EDP for a demand P_d can be formalized as

$$\min_{P_1,\dots,P_m} \quad \sum_{i=1}^{m} J_i(P_i) \tag{7a}$$

$$\text{subject to:} \quad \sum_{i=1}^{m} P_i - P_\ell(\vec{P}) \geq P_d \tag{7b}$$

$$P_i \in \mathcal{P}_i i = 1, \dots, m, \tag{7c}$$

with $J_i(P_i)$, $P_\ell(\vec{P})$, and $\mathcal{P}_i$ respectively defined in (1), (5), and (6). Problem (7) has a quadratic cost (7a), a quadratic *global* constraint (7b) involving the decision variables of all generators, and m *local* constraints (7c) each involving the power of a single generator. Since the local set $\mathcal{P}_i$ is the intersection between an interval and a union of disjoint intervals arising from commission/decommission and prohibited zones, (7) is a non-convex problem, which is difficult to solve.

In this letter, we leverage Lagrangian duality along with the scalar nature of the coupling constraint and propose a computationally efficient bisection-based algorithm to provide a feasible (possibly suboptimal) solution to (7).

III. Proposed Approach

Upon a close inspection of (7), it is clear that penalizing the violation of constraint (7b) instead of enforcing it as a constraint would turn the problem from a quadratically constrained non-convex program into a program that is still non-convex but with a quadratic cost function only. Moreover, in such a case, the P_i's would be independently constrained by the sets $\mathcal{P}_i$ and would be coupled through the (quadratic) cost function only. This is the typical situation in which Lagrangian duality can help in dealing with the complicating constraint (7b). Let us therefore lift (7b) to the cost function through a (single) Lagrange multiplier $\lambda \geq 0$, define the Lagrangian function

$$\mathcal{L}(\vec{P}, \lambda) = \sum_{i=1}^{m} J_i(P_i) + \lambda\Big(P_d + P_\ell(\vec{P}) - \sum_{i=1}^{m} P_i\Big), \tag{8}$$

construct the dual function[1]

$$\varphi(\lambda) = \min_{\{P_i \in \mathcal{P}_i\}_{i=1}^m} \mathcal{L}(\vec{P}, \lambda), \tag{9}$$

and pose the dual problem

$$\max_{\lambda \geq 0} \varphi(\lambda). \tag{10}$$

The dual problem in (10) is convex, despite (7) is non-convex, and its optimal cost $\varphi^\star$ provides a lower bound to the optimal cost $J^\star$ of (7), see e.g., [18, Sec. 5.1.3]. Moreover, the scalar nature of constraint (7b) makes the dual function $\varphi(\lambda)$ one-dimensional and, thus, easy to maximize, as the optimal value $\lambda^\star$ can be found by looking for a zero of the sub-differential map of the dual function. One could thus compute an optimal dual solution $\lambda^\star$, and, then, recover a primal solution by minimizing the Lagrangian at $\lambda = \lambda^\star$:

$$\big[P_1(\lambda^\star) \cdots P_m(\lambda^\star)\big]^\top = \operatorname*{arg\,min}_{\{P_i \in \mathcal{P}_i\}_{i=1}^m} \mathcal{L}(\vec{P}, \lambda^\star). \tag{11}$$

Such a solution, however, is not guaranteed to satisfy the aggregate power demand since the dualized constraint (7b) is not directly enforced in (11).

The approach proposed in this letter overcomes such limitation, resorting to the dual bisection method introduced in [10] for general non-convex problems with a single complicating constraint. The procedure is guaranteed to either converge to an optimal primal solution in a finite number of iterations or generate a sequence of feasible primal solutions with non-increasing cost. We will first describe the proposed scheme (reported in Algorithm 1) and its properties, which will then be theoretically discussed in the next subsection.

The bisection procedure starts with an interval $[\underline{\lambda}, \bar{\lambda}] = [0, \lambda_{\text{start}}]$ (cf. Steps 1 and 2), where

$$\lambda_{\text{start}} = \frac{\sum_{i=1}^m \big(J_i(\tilde{P}_i^{\max}) - \min_{P_i \in \mathcal{P}_i} J_i(P_i)\big)}{\sum_{i=1}^m \tilde{P}_i^{\max} - P_\ell(\vec{\tilde{P}}^{\max}) - P_d} \tag{12}$$

is selected to ensure that $\lambda^\star \in [0, \lambda_{\text{start}}]$, as later discussed in Section III-A. Note that λ_{start} is not difficult to compute as the numerator is the difference between the production costs associated to the maximum admissible power generation and the costs associated to the minimum power generation irrespectively of demand satisfaction summed across all generators, while the denominator is the difference between the maximum admissible power generation (minus losses) and the actual power demand. Then, a first *feasible* power allocation $\vec{P}$ is computed by minimizing the Lagrangian in (8) with a penalization coefficient for constraint (7b) equal to $\lambda = \bar{\lambda} = \lambda_{\text{start}}$ (cf. Step 3). If such allocation matches the demand P_d exactly (cf. Step 4), then it is also *optimal* and is readily returned in Step 5, otherwise is saved as the current best

[1]Note that function $\varphi(\cdot)$ is well defined since the sets $\mathcal{P}_i$, $i = 1, \ldots, m$ are closed and bounded, and functions $J_i(\cdot)$ and $P_\ell(\cdot)$ are continuous.

Algorithm 1 Bisect EDP

1: $\underline{\lambda} \leftarrow 0$
2: $\bar{\lambda} \leftarrow \lambda_{\text{start}}$ in (12)
3: $\vec{P} \leftarrow \arg\min_{\{P_i \in \mathcal{P}_i\}_{i=1}^m} \mathcal{L}(\vec{P}, \bar{\lambda})$
4: **if** $\sum_{i=1}^m P_i - P_\ell(\vec{P}) = P_d$ **then**
5:     **return** $\vec{P}$
6: **end if**
7: $\vec{P}_{\text{best}} \leftarrow \vec{P}$
8: **repeat**
9:     $\hat{\lambda} \leftarrow \frac{1}{2}(\bar{\lambda} + \underline{\lambda})$
10:     $\vec{P} \leftarrow \arg\min_{\{P_i \in \mathcal{P}_i\}_{i=1}^m} \mathcal{L}(\vec{P}, \hat{\lambda})$
11:     **if** $\sum_{i=1}^m P_i - P_\ell(\vec{P}) = P_d$ **then**
12:         **return** $\vec{P}$
13:     **else if** $\sum_{i=1}^m P_i - P_\ell(\vec{P}) > P_d$ **then**
14:         $\vec{P}^{\text{best}} \leftarrow \vec{P}$
15:         $\bar{\lambda} \leftarrow \hat{\lambda}$
16:     **else if** $\sum_{i=1}^m P_i - P_\ell(\vec{P}) < P_d$ **then**
17:         $\underline{\lambda} \leftarrow \hat{\lambda}$
18:     **end if**
19: **until** some stopping criterion is met
20: **return** $\vec{P}_{\text{best}}$

allocation $\vec{P}_{\text{best}}$ in Step 7 and the algorithm proceeds to the bisection loop (cf. Steps 8-19).

At the beginning of each bisection iteration the midpoint $\hat{\lambda}$ of the interval $[\underline{\lambda}, \bar{\lambda}]$ is computed (cf. Step 9) and a new allocation $\vec{P}$ is obtained by minimizing the Lagrangian in (8) with a penalization coefficient for constraint (7b) equal to $\lambda = \hat{\lambda}$ (cf. Step 10). If such allocation matches the demand P_d exactly (cf. Step 11), then it is also *optimal* and is readily returned in Step 12. If, instead, $\vec{P}$ strictly satisfies the demand (cf. Step 13), then it is *feasible* for (7) and its cost is no-worse than that of the current $\vec{P}_{\text{best}}$, since $\hat{\lambda} < \bar{\lambda}$ and, hence, constraint (7b) has been penalized less in favor of a better cost. Therefore, $\vec{P}$ is selected as the new best allocation and saved into $\vec{P}_{\text{best}}$ in Step 14. At the same time, since we are producing strictly more than the demand P_d, it may be that we are still over-penalizing constraint (7b), and, hence, $\hat{\lambda}$ is selected as the new upper extreme of the bisection interval in Step 15. If, instead, $\vec{P}$ is not enough to satisfy the demand (cf. Step 16), then it is *infeasible* for (7) and it is discarded. Accordingly, since we are producing strictly less than the demand P_d, we are under-penalizing constraint (7b), and, hence, $\hat{\lambda}$ is selected as the new lower extreme of the bisection interval in Step 17. The loop continues until some stopping criterion is met, like when a maximum number of iterations is reached or when the length of the interval $[\underline{\lambda}, \bar{\lambda}]$ falls below a certain threshold. Whenever the loop stops, the algorithm returns the best allocation found $\vec{P}_{\text{best}}$ (cf. Step 20).

A. Theoretical Discussion

In this section we show that problem (7) fits the framework proposed in [10] and, hence, the DualBi algorithm proposed in [10] can be applied and it actually reduces to Algorithm 1 when applied to problem (7). Any claim made in the previous section will thus be justified by the corresponding claim in [10]. Let us start by noting that problem (7) has the structure

$$\min_{x\in X} \quad f(x) \tag{13a}$$
$$\text{subject to:} \quad v(x) \le 0 \tag{13b}$$

of [10, $\mathcal{P}$] simply defining the x, $f(x)$, $v(x)$, and X quantities in [10] as

$$x = \vec{P} \tag{14a}$$
$$f(x) = \sum_{i=1}^{m} J_i(P_i), \tag{14b}$$
$$v(x) = P_d + P_\ell(\vec{P}) - \sum_{i=1}^{m} P_i, \tag{14c}$$
$$X = \mathcal{P}_1 \times \cdots \times \mathcal{P}_m. \tag{14d}$$

The Lagrangian in (8) can be equivalently expressed as $\mathcal{L}(x, \lambda) = f(x) + \lambda v(x)$ and the dual function as $\varphi(\lambda) = \min_{x\in X} \mathcal{L}(x, \lambda)$.

As we discussed in the previous section, $v(x) \le 0$ (i.e., (7b)) is indeed a complicating constraint and is also a support constraint, as the demand P_d will not be satisfied by setting the generators at their minimum admissible power. Since the sets $\mathcal{P}_i$, $i = 1, \dots, m$ are all compact, X is compact too. Moreover, since $J_i(P_i)$ is quadratic in P_i for all $i = 1, \dots, m$ and $P_\ell(\vec{P})$ is quadratic in $P_1, \dots, P_m$, both $f(x)$ and $v(x)$ are scalar continuous functions of $x \in X$.

Since we assumed $P_d < \sum_{i=1}^{m} \bar{P}_i^{\max} - P_\ell(\vec{\bar{P}}^{\max})$, this is equivalent to know that there exists an $\tilde{x} = \vec{\bar{P}}^{\max}$ such that $v(\tilde{x}) < 0$, meaning that

$$\varphi(\lambda) = \min_{x\in X} \mathcal{L}(x, \lambda) = \min_{x\in X} f(x) + \lambda v(x) \le f(\tilde{x}) + \lambda v(\tilde{x}).$$

This implies that $\limsup_{\lambda\to+\infty} \varphi(\lambda) = -\infty$ and, since $\varphi(\cdot)$ is concave, that the level sets of $\varphi(\lambda)$ are compact, which, together with continuity, ensures that an optimal dual solution $\lambda^\star$ exists and, hence, [10, Assumption 1] is satisfied. Moreover, according to [10, Th. 2], setting

$$\lambda_{\text{start}} = \frac{\varphi(0) - f(\tilde{x})}{v(\tilde{x})} = \frac{\min_{x\in X} \mathcal{L}(x, 0) - f(\tilde{x})}{v(\tilde{x})} \overset{(14)}{\equiv} (12),$$

results in DualBi skipping [10, Steps 4–10 in Algorithm 1] and executing only [10, Steps 1–3 and Steps 11–30 in Algorithm 1], which are equivalent to the proposed Algorithm 1 given the identifications in (14). Therefore, by [10, Th. 1], Algorithm 1 either returns an optimal solution to (7) after a finite number of iterations or refines its power allocation $\vec{P}_{\text{best}}$ by progressively reducing its overall cost.

B. Implementation

Running Algorithm 1 requires solving problem (9) at different values of the dual variable λ. Despite being simpler than (7), it still requires minimizing a quadratic cost function over a non-convex set. Luckily each P_i is independently constrained to belong to a union of intervals, which can be easily reformulated as mixed-integer linear constraints.

For each generator i, let $\sigma_{i,j} \in \{0, 1\}$, $j = 1, \dots, N_i$ be additional binary decision variables encoding whether P_i belongs to the j-th allowed power interval ($\sigma_{i,j} = 1$) or not ($\sigma_{i,j} = 0$).

Then, the mutually exclusive conditions in (2) can be equivalently reformulated as

$$\sum_{j=1}^{N_i} \sigma_{i,j} \underline{P}_{i,j} \le P_i \le \sum_{j=1}^{N_i} \sigma_{i,j} \bar{P}_{i,j} \tag{15a}$$
$$\sum_{j=1}^{N_i} \sigma_{i,j} \le 1, \tag{15b}$$

where (15b) constrain P_i to belong to at most one interval, while (15a) specifies the interval extremes based on the values of $\sigma_{i,1}, \dots, \sigma_{i,N_i}$ (cf. (2b)). When $\sigma_{i,1} = \cdots = \sigma_{i,N_i} = 0$, which is allowed by (15b), constraint (15a) enforces $P_i = 0$ (cf. (2a)).

This shows that each local set $\mathcal{P}_i$ can be reformulated as

$$\mathcal{P}_i = \{P_i \in \mathbb{R} \colon \exists\, \sigma_{i,1}, \dots, \sigma_{i,N_i} \in \{0, 1\} \colon (3) \wedge (15)\},$$

which is a mixed-integer set described by linear inequalities, easily handled by off-the-shelf solvers. Moreover, this also shows that problem (7) can be posed as a Mixed-Integer Quadratically Constrained Quadratic Program (MIQCQP), whereas the problem solved at each iteration of Algorithm 1 is (9), which can be posed as a significantly simpler Mixed-Integer Quadratic Program (MIQP).

C. Computational Complexity

The MIQCQP (7) has a combinatorial complexity that scales exponentially in the number m of generators and depends on the number K of disjoint convex sets in $\mathcal{P}_1 \times \cdots \times \mathcal{P}_m$, which is given by $K = \prod_{i=1}^{m} n_{\mathcal{P}_i}$, with $n_{\mathcal{P}_i} \le N_i + 1$ number of disjoint intervals in $\mathcal{P}_i$ in (6). To solve (7), one could, in principle, proceed via enumeration by fixing a power interval in $\mathcal{P}_i$ for each generator i, solving the resulting (convex) Quadratically Constrained Quadratic Program (QCQP), and then choosing the best among the obtained K solutions. By lifting the quadratic constraint to the cost, the approach in this letter requires solving the MIQP (9), instead of the MIQCQP (7), at each iteration of Algorithm 1. Admittedly, (9) has the same combinatorial nature of (7). However, its solution by enumeration would involve K (much simpler) quadratic programs with box constraints instead of K QCQPs and, as m grows, this balances the fact that it needs to be solved repeatedly (see the results reported in Section IV).

IV. Numerical Simulations

We now assess the efficacy of the proposed approach on two benchmark EDPs described in [6, Section V-A], and then test its scalability on randomly generated problem instances of different dimensions. All tests are performed on a laptop equipped with an Intel Core i7-9750HF CPU @2.60GHz and 16GB of RAM. Algorithm 1 is implemented in MATLAB R2020b and uses CPLEX v12.10 to solve the MIQP obtained at each iteration by lifting the complicating constraint. In each test, we let Algorithm 1 run until the length of the interval $[\underline{\lambda}, \ \bar{\lambda}]$ falls below 10^{-8}.

We measure the quality of a solution $\vec{P}$ of (7) based on its normalized excess of production

$$E(\vec{P}) = \frac{\sum_{i=1}^{m} P_i - P_\ell(\vec{P}) - P_d}{P_d}$$

TABLE I

IEEE Benchmark 1: Comparison of the Solution Obtained via Algorithm 1 With Those Obtained by Alternative Resolution Methods in the Literature. The Solution Marked With $*$ Violates the Aggregate Demand Constraint and Is Super-Optimal

Unit	GA [5]	PSO [6]	Bi-B&B [8]	DE [15]	Algorithm 1
P_1	474.81	447.50	447.40	448.27	447.08
P_2	178.64	173.32	173.24	172.96	173.19
P_3	262.21	263.47	263.38	263.44	263.93
P_4	134.28	139.06	138.98	139.30	139.06
P_5	151.90	165.48	165.39	165.28	165.58
P_6	74.18	87.13	87.05	86.80	86.63
$\Delta_J(\cdot)$	$\mathbf{1.02 \cdot 10^{-3}}$	$\mathbf{4.18 \cdot 10^{-4}}$	$\mathbf{-2.59 \cdot 10^{-5*}}$	$\mathbf{3.99 \cdot 10^{-4}}$	$\mathbf{1.47 \cdot 10^{-9}}$
$E(\cdot)$	$3.71 \cdot 10^{-4}$	$3.77 \cdot 10^{-4}$	$-2.38 \cdot 10^{-5*}$	$3.60 \cdot 10^{-4}$	$1.48 \cdot 10^{-9}$

and its relative optimality gap

$$\Delta_J(\vec{P}) = \frac{J(\vec{P}) - J^\star}{J^\star} \quad (16)$$

where $J(\vec{P}) = \sum_{i=1}^{m} J(P_i)$ and $J^\star$ is the optimal cost, determined by solving problem (7) via Gurobi, setting its optimality and feasibility tolerances to 10^{-8}.

A. Benchmark Examples

The first benchmark problem considers an IEEE case-study introduced in [6] with $m = 6$ generators that must satisfy a demand of 1263 MW. The parameters of the system and loss coefficients are reported in [6, Table I-II and Appendix] and are omitted for the sake of conciseness. Table I compares the best feasible solution $\vec{P}^{\text{best}}$ obtained by Algorithm 1 with the solutions computed by the Genetic Algorithm in [5], the Particle Swarm Optimization method in [6], the bi-level Branch-and-Bound approach in [8], and the decentralized scheme in [15]. Note that the solution obtained via the scheme in [8] has a negative relative optimality gap and a negative normalized excess of production. This means that it is super-optimal but unfeasible. Indeed, it results in a power loss of 12.47 MW (as opposed to the 12.44 reported in [8]) and in a power production that does not satisfy the total power demand. The solution is marked with an $*$ in the table and is excluded from the discussion in the sequel. Results show that Algorithm 1 outperforms its competitors. The normalized excess of production $E(\vec{P})$ is of the order of 10^{-9}, which implies that the allocation practically satisfies (7) with equality. In addition, the relative optimality gap of the solution $\vec{P}^{\text{best}}$ is at least five orders of magnitude below the one achieved by the other methods and is smaller than the optimality tolerance chosen for the Gurobi solver, meaning that the obtained solution is also optimal.

The second benchmark test is another IEEE case study first addressed in [6] comprising a larger number of generator units (15 instead of 6) to satisfy a power demand equal to 2630 MW. Only 4 of the generators present prohibited operating zones (see [6, Tables V-VI and Appendix] for the parameters). Also in this case Algorithm 1 finds an optimal solution given that its relative optimality gap is equal to $2.42 \cdot 10^{-10}$, with an excess of production of $1.08 \cdot 10^{-6}$.

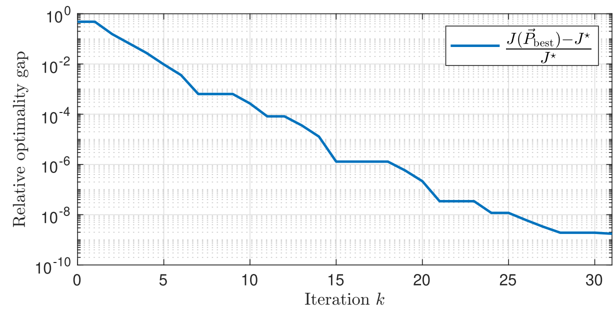

Fig. 1. EDP with $m = 50$ generators: relative optimality gap of the solution computed by Algorithm 1 as a function of the iteration number.

B. Artificially Generated EDPs

We now assess the effectiveness of the proposed procedure on randomly generated yet realistic instances of (7) (see the protocol in the Appendix).

Figure 1 shows the evolution of the relative optimality gap of the solution $\vec{P}_{\text{best}}$ computed by Algorithm 1 throughout the iterations for an instance of the EDP (7) with $m = 50$ generators. As expected, the performance attained by $\vec{P}_{\text{best}}$ improves throughout the iterations, and tends monotonically to 0 as the number of iterations grows.

If we run Algorithm 1 on 100 instances of problem (7) with $m = 50$ generators and different parameter sets, then, it successfully computes a feasible solution for each test and in 76% of the cases the relative optimality gap Δ_J is below the optimality tolerance of 10^{-8} chosen for the solver, meaning that the obtained solution is also optimal for (7). In the remaining tests, the average relative optimality gap is $3.73 \cdot 10^{-3}$ and, thus, the obtained solution is still close to the optimum.

We now consider instances of problem (7) with an increasing number m of generators. The first two columns of Table II report the normalized excess of production and the relative optimality gap attained by the solution $\vec{P}^{\text{best}}$ computed by Algorithm 1 for different values of m. The relative optimality gap never exceeds 0.031% of the optimal cost and the extra-production is always below 0.029% of the power demand.

The third and fourth columns of Table II compare the execution time of Gurobi and Algorithm 1, respectively. As the number of generators m increases, both algorithms exhibit a rise in the computational time, reflecting an increase of the problem complexity. However, the proposed approach scales better with the size of the problem and is able to find feasible

TABLE II
PERFORMANCE OF ALGORITHM 1 AS A FUNCTION OF m: RELATIVE OPTIMALITY GAP, NORMALIZED EXCESS OF PRODUCTION, TIME REQUIRED TO COMPUTE THE OPTIMAL SOLUTION, EXECUTION TIME

m	$\Delta_J(\vec{P}^{\text{best}})$	$E(\vec{P}^{\text{best}})$	t_G [s]	t_{A1} [s]
50	$5.0 \cdot 10^{-10}$	$4.0 \cdot 10^{-10}$	0.8	2.65
100	$-5.2 \cdot 10^{-6}$	$2.6 \cdot 10^{-10}$	2.6	4.93
150	$7.1 \cdot 10^{-6}$	$6.0 \cdot 10^{-10}$	38.0	7.30
250	$-5.5 \cdot 10^{-5}$	$3.3 \cdot 10^{-7}$	1689.2	18.03
500	$3.1 \cdot 10^{-4}$	$2.9 \cdot 10^{-4}$	2020.4	85.87
1000	$8.4 \cdot 10^{-5}$	$8.3 \cdot 10^{-5}$	> 18000	789.07
1500	$2.3 \cdot 10^{-5}$	$2.8 \cdot 10^{-5}$	> 18000	3781.7
2000	$8.8 \cdot 10^{-5}$	$8.7 \cdot 10^{-5}$	> 18000	8900.7

solutions for instances with up to $m = 2000$ generators in less than 3 hours using the available computing resources.

Conversely, for instances with $m \geq 1000$ Gurobi is not able to compute an optimal solution within 5 hours. For all those instances, the relative optimality gap Δ_J achieved by Algorithm 1 is computed using in (16) the optimal dual cost $\varphi^\star$ in place of $J^\star$, which increases the index value since $\varphi^\star$ is a lower bound of $J^\star$.

V. CONCLUSION

We addressed economic dispatch problems accounting for power losses and prohibited operating zones. We proposed a duality-based approach that recovers computational tractability by dualizing the aggregate power demand constraint and computes a feasible solution via a dual bisection iterative algorithm. Simulations on benchmark examples showcase the superiority of the approach with respect to state-of-the-art competitors, which provide solutions that either yield a higher production cost or do not meet the aggregate power demand. Tests on randomly generated EDP instances with increasing size show that the proposed approach is scalable and able to compute near-optimal solutions.

APPENDIX

GENERATION OF AN EDP WITH m UNITS

The minimum and maximum power $P_i^{\min}$, $P_i^{\max}$ of each generator are multiples of 5 MW selected at random in the interval [10, 300] MW and $[1.1P_i^{\min},\ 1.6P_i^{\min}]$, respectively. We set the number of generators characterized by prohibited power zones to $\lceil 0.2m \rceil$. For each of these generators, the number of operating zones, their length and position within the allowed power production range are selected at random so that the different zones do not intersect and keep the problem feasible. The upper and lower ramp limits $\underline{\Delta}_i$ and $\bar{\Delta}_i$ are multiple of 5 MW randomly extracted from $[0, P_i^{\max} - P_i^{\min}]$. The power $P_{i,0}$ produced in the previous time slot is selected uniformly within one of the feasible intervals $[\underline{P}_{i,j^0}, \bar{P}_{i,j^0}]$ for some j^0 extracted uniformly from $\{1, \ldots, N_i\}$. To ensure that the $m \times m$ matrix containing the loss coefficients b_{ij} is positive definite and, thus, defines an actual power loss, we set it equal to the sample covariance matrix of n_{vec} realizations of an m-dimensional multivariate normal random variable vector with unitary mean and variance equal to 10^{-4}. The number of observations n_{vec} is set to 100 and is doubled until the smallest (positive) eigenvalue of the resulting covariance matrix is larger than 10^{-5}. The coefficients b_{i0} and b_{00} are extracted uniformly from the intervals $[10^{-8},\ 10^{-5}]$ and $[10^{-4},\ 10^{-2}]$, respectively. The cost coefficients are selected at random within the following intervals $\omega_{i0} \in [0,\ 550]$, $\omega_{i1} \in [5,\ 15]$ and $\omega_{i2} \in [0,\ 4 \cdot 10^{-3}]$. The total power demand P_d is set equal to a percentage ρ of the maximum output power that the aggregate can produce, with ρ extracted uniformly from the interval [20%, 80%]. Unfeasible instances are discarded and replaced by feasible ones. This protocol allows to create parameter sets comparable to the IEEE benchmark examples in [6] and, thus, reasonable for a real application.

REFERENCES

[1] R. H. Kerr, J. L. Scheidt, A. J. Fontanna, and J. K. Wiley, "Unit commitment," *IEEE Trans. Power App. Syst.*, vol. PAS-85, no. 5, pp. 417–421, May 1966.
[2] B. Chowdhury and S. Rahman, "A review of recent advances in economic dispatch," *IEEE Trans. Power Syst.*, vol. 5, no. 4, pp. 1248–1259, Nov. 1990.
[3] H. Saadat, *Power Systems Analysis*, 3rd ed. New York, NY, USA: McGraw-Hill, 2010.
[4] F. Marzbani and A. Abdelfatah, "Economic dispatch optimization strategies and problem formulation: A comprehensive review," *Energies*, vol. 17, no. 3, p. 550, 2024.
[5] P.-H. Chen and H.-C. Chang, "Large-scale economic dispatch by genetic algorithm," *IEEE Trans. Power Syst.*, vol. 10, no. 4, pp. 1919–1926, Nov. 1995.
[6] Z. L. Gaing, "Particle swarm optimization to solving the economic dispatch considering the generator constraints," *IEEE Trans. Power Syst.*, vol. 18, no. 3, pp. 1187–1195, Aug. 2003.
[7] W. W. Lee and M. R. B. Hashim, "A hybrid algorithm based on artificial bee colony and artificial rabbits optimization for solving economic dispatch problem," in *Proc. IEEE Int. Conf. Automat. Control Intell. Syst. (I2CACIS)*, 2023, pp. 298–303.
[8] T. Ding, R. Bo, F. Li, and H. Sun, "A bi-level branch and bound method for economic dispatch with disjoint prohibited zones considering network losses," *IEEE Trans. Power Syst.*, vol. 30, no. 6, pp. 2841–2855, Nov. 2015.
[9] S. Pan, J. Jian, H. Chen, and L. Yang, "A full mixed-integer linear programming formulation for economic dispatch with valve-point effects, transmission loss and prohibited operating zones," *Elect. Power Syst. Res.*, vol. 180, Mar. 2020, Art. no. 106061.
[10] L. Manieri, A. Falsone, and M. Prandini, "DualBi: A dual bisection algorithm for non-convex problems with a scalar complicating constraint," 2024, *arXiv:2402.03013*.
[11] M. Garcia and R. Baldick, "Approximating economic dispatch by linearizing transmission losses," *IEEE Trans. Power Syst.*, vol. 35, no. 2, pp. 1009–1022, Mar. 2020.
[12] S. O. Orero and M. R. Irving, "Economic dispatch of generators with prohibited operating zones: A genetic algorithm approach," *Inst. Eng. Technol. Proc.-Gener., Transm. Distribut.*, vol. 143, no. 6, pp. 529–534, 1996.
[13] A. I. Selvakumar and K. Thanushkodi, "Anti-predatory particle swarm optimization: Solution to nonconvex economic dispatch problems," *Elect. Power Syst. Res.*, vol. 78, no. 1, pp. 2–10, 2008.
[14] A. Pereira-Neto, C. Unsihuay, and O. Saavedra, "Efficient evolutionary strategy optimisation procedure to solve the nonconvex economic dispatch problem with generator constraints," *Inst. Eng. Technol. Proc.-Gener., Transm. Distribut.*, vol. 152, no. 5, pp. 653–660, 2005.
[15] W. T. Elsayed and E. F. El-Saadany, "A fully Decentralized approach for solving the economic dispatch problem," *IEEE Trans. Power Syst.*, vol. 30, no. 4, pp. 2179–2189, Jul. 2015.
[16] A. Carrillo-Galvez, F. Flores-Bazan, and E. López, "A duality theory approach to the environmental/economic dispatch problem," *Elect. Power Syst. Res.*, vol. 184, Jul. 2020, Art. no. 106285.
[17] L. G. Papageorgiou and E. S. Fraga, "A mixed integer quadratic programming formulation for the economic dispatch of generators with prohibited operating zones," *Elect. Power Syst. Res.*, vol. 77, no. 10, pp. 1292–1296, 2007.
[18] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

Open Access funding provided by 'Politecnico di Milano' within the CRUI CARE Agreement