An MISOCP-Based Decomposition Approach for the Unit Commitment Problem With AC Power Flows

Deniz Tuncer and Burak Kocuk

Abstract—Unit Commitment (UC) and Optimal Power Flow (OPF) are two fundamental problems in short-term electric power systems planning that are traditionally solved sequentially. The state-of-the-art mostly uses a direct current (DC) approximation of the power flow equations. However, utilizing the DC approach in the UC-level may lead to infeasible or suboptimal generator commitment schedules for the OPF problem. In this paper, we aim to simultaneously solve the UC Problem with alternating current (AC) power flow equations, which combines the challenging nature of both UC and OPF Problems. Due to the highly nonconvex nature of the AC flow equations, we utilize the mixed-integer second-order cone programming (MISOCP) relaxation of the UC Problem as the basis of our solution approach. The MISOCP relaxation is utilized for finding both a lower bound and a candidate generator commitment schedule. Once this schedule is obtained, we solve a multi-period OPF problem to obtain feasible solutions for the UC problem with AC power flows. For smaller instances, we develop two different algorithms that exploit the recent advances in the OPF literature and obtain high-quality feasible solutions with provably small optimality gaps. For solving larger instances, we develop a Lagrangian decomposition based approach that yields promising results.

Index Terms—Optimal power flow, unit commitment, mixed-integer programming, nonlinear programming, second-order cone programming.

NOMENCLATURE

Decision Variables

\[
|V_{i,t}|, \quad \theta_{i,t}, \quad p_{i,t}^d(q_{i,t}^d)
\]

Voltage magnitude of bus \(i\) in period \(t\).
Voltage phase angle of bus \(i\) in period \(t\).
Active (reactive) power generation amount of generator \(i\) in period \(t\).
Active (reactive) power through line \((i,j)\) in period \(t\) in forward direction.
Active (reactive) power through line \((i,j)\) in period \(t\) in backward direction.
Commitment status of generator \(i\) in period \(t\).
Start up status of generator \(i\) in period \(t\).
Start up status of generator \(i\) in period \(t\).

Parameters

\[
 f_{ij,t}, \quad \delta(i), \quad g_{ij}(i) \quad \lambda_{r}^{u}, \quad \lambda_{r}^{l}, \quad \lambda_{d}^{u}, \quad \lambda_{d}^{l}, \quad \lambda_{u}^{r}, \quad \lambda_{u}^{l}
\]

Real power flow across the line \((i,j)\).
Active (reactive) energy demand of bus \(i\) in period \(t\).
Set of neighbors of bus \(i\).
Shunt conductance (susceptance) of bus \(i\).
Minimum (maximum) active generation limit of generator \(i\).
Minimum (maximum) reactive generation limit of generator \(i\).
Minimum (maximum) voltage magnitude of bus \(i\).
Lower (upper) bound on the phase angle of line \((i,j)\).
Admittance matrix of line \((i,j)\).
Real and imaginary parts of \(Y_{ij}^{r}\).
Real and imaginary parts of \(Y_{ij}^{l}\).
Real and imaginary parts of \(Y_{ij}^{l}\).
Maximum power that can pass through line \((i,j)\).
Operational cost of generator \(i\) in period \(t\).
Ramp up (down) rate of generator \(i\).
Minimum uptime (downtime) of generator \(i\).
Fixed cost of generator \(i\) in period \(t\).
Lagrangian multiplier for minimum uptime constraint (9b).
Lagrangian multiplier for minimum downtime constraint (9c).
Lagrangian multiplier for the logical constraint (9d).
Lagrangian multiplier for the ramp down constraint (9e).
Lagrangian multiplier for the ramp up constraint (9f).

I. INTRODUCTION

The Optimal Power Flow (OPF) and the Unit Commitment (UC) Problems are the backbones of the short-term power system planning. The objective of the UC Problem is to determine the most cost-efficient schedule of generators for the next day whereas the objective of the OPF Problem is to find a...
minimum cost feasible electricity dispatch in a power network
given the generator schedules. The UC Problem is typically
solved with the approximate direct current (DC) power flow
equations whereas the OPF Problem requires the AC power flow
equations to be taken into account. Although it is common to
solve these two problems sequentially, simultaneously deciding
the generator schedules and power dispatch can yield lower total
cost. This is the main motivation and the goal of our paper.

Since OPF is a fundamental problem, the related literature
is quite rich. There are various methods for solving the OPF
Problem, which we categorize into three groups: i) nonlinear
programming (NLP) methods that aim to locate locally
optimal solutions, ii) linear programming (LP) methods that
approximately solve the problem, iii) conic relaxations based on
semidefinite programming (SDP) relaxations and second order
cone programming (SOCP) that can provide dual bounds.

The first group of methods used in the OPF literature is the
NLP based methods. They generally focus on obtaining the
stationary points of the problem, utilizing Newton’s method
or interior point methods. Newton’s method requires using La-
grangian multipliers as penalty terms for the constraints in the
constrained optimization problems [1]. The papers [2] and [3]
utilize the Newton’s method for solving the OPF Problem. By
transforming the optimality conditions, [4] makes the problem
solvable by an algorithm solely based on the Newton-Raphson
method. While these methods are good at finding stationary
points, they may get stuck at a local optimum point since the
OPF Problem is nonconvex [5]. Also, since the initial point
is important for these methods they may fail to converge to a
solution [6].

The second group of algorithms in the OPF literature uses LP
based methods, which utilize the DC approximation of the AC
power flows [7], [8], [9]. Since LPs can be solved efficiently, LP
based methods are widely used in industry to solve the OPF Prob-
lem. LP methods are also used as a subroutine in a successive
linear programming framework [10]. However, the LP methods
may not produce an AC feasible dispatch since the reactive
powers and losses are completely ignored or approximated.

The third group of algorithms in the OPF literature uti-
lizes conic programming relaxations. In particular, SDP relax-
ations [11], [12] have drawn significant interest due to their
strength and polynomial solvability. Such relaxations can
provide globally optimal solutions when the relaxation is exact,
although this can be only guaranteed for very special cases [13].
Penalized SDP problems are also proposed to find approximately
feasible solutions [14]. The computational burden of solving
SDP relaxations can be relieved by cheaper but weaker relax-
ations in the form of the SOCP relaxation [15]. The authors
of [16] exploit the computational advantages of the SOCP re-
laxation and strengthen it using three types of inequalities so
that their approach is both more efficient and accurate than the
standard SDP relaxation. However, despite all the effort, conic
programming relaxations are not guaranteed to provide feasible
solutions and they are very useful for dual bounding purposes.

We have mentioned the rich literature on the OPF Prob-
lem. However, the literature on the Multi-period OPF (MOPF)
Problem, which can be seen as a generalization of the OPF
Problem and a special case of the UC Problem, is somewhat
limited. A generalized Benders decomposition is used in [17],
in which network constraints are modeled according to a DC
approximation. An LP based interior point algorithm is used in
[18], where Newton-Raphson method is used as a corrector.
The problem is formulated as a two stage nonlinear program
in [19], followed by solving the problem with an interior point
method.

The literature on the UC Problem is also quite rich. Over the
years, many different approaches for the problem are developed
which we categorized into four groups: i) Lagrangian relaxation,
ii) mixed-integer linear programming (MILP), iii) Benders de-
composition, and iv) conic relaxations.

The first method we would like to mention for the UC Prob-
lem is the Lagrangian Relaxation (LR) method, which aims to
decompose it into smaller subproblems. A branch-and-bound
algorithm is proposed in [20] where they use a Lagrangian
method and they decompose the problem into problems with
a single generator. LR is utilized in [21] in order to find a
lower bound and a primal relaxed solution, where the solution
is used in a heuristic resolution method. The main drawback of
these methods is that the solution obtained by solving LR dual
problems might not be primal feasible for the UC Problem [22].

The second method, mixed integer linear program (MILP)
based methods depend on the DC approximation of the power
flow equations and is the state-of-the-art for the UC Problem.
For tightening the MILP formulation, different methods and
formulations are proposed in [23], [24], [25], [26]. The drawback
of this method is that since it disregards the reactive powers,
the solutions might cause stress in the power network. One exception
in this direction is [27], which incorporates AC power flow
constraints are incorporated into the MILP model as piecewise
linear constraints.

The third method we would like to mention is the Benders
Decomposition (BD) [28], [29]. These references have a master
problem and nonlinear subproblems, from which they obtain
Benders cuts for the master problem. The drawback of the BD
approach is that it converges slowly and is computationally
expensive [30].

There is only very limited literature on the UC Problem
with AC power flow equations and this literature typically
utilize conic relaxation methods such as SDP relaxation [12]. A
strengthened SDP problem is solved in [31], which is a convex
problem and its minimum can be found efficiently. The SDP
relaxation together with Benders’ decomposition is considered in
[32]. A mixed-integer second-order cone programming (MISO-
COP) master problem with MOPF subproblems are considered in
[33], [34]. Although this group of methods that utilize conic
relaxations seem to be the most promising approaches to solve
the UC Problem to global optimality, they mostly rely on ex-
ensive optimization problems to handle nonconvexity (such as
SDPs and MILP discretizations). In addition, this line of research
does not exploit the recent advances in the convexification of the
AC OPF Problem fully. Our paper contributes to this literature
by proposing relatively cheap and strengthened MISOCOCP based
methods that can provide provably high quality feasible solu-
tions within a reasonable computational budget.
In our study, we aim to solve the UC Problem with AC power flows from a global optimization perspective. We utilize the mixed-integer conic relaxation of the UC Problem and solve an MISOCP in order to obtain a candidate generator commitment schedule and also a lower bound for the problem. Then, given the commitment schedule, we solve an MOPF Problem and find an AC feasible solution using a local solver. The lower bounds are compared with the upper bounds obtained from the feasible solutions, and we are able to provide an optimality gap.

For solving the UC Problem with AC power flows, we propose three different MISOCP-based algorithms. The first algorithm we propose solves an MISOCP, followed by an NLP problem. In the second algorithm, we strengthen the MISOCP problem by utilizing two methods from [16]. These algorithms enable us to solve the UC Problem with AC power flows efficiently, and within 1.3% optimality gap for small-size instances. However, for large-scale instances solving a 24-period MISOCP problem is quite challenging for which we also develop a novel decomposition algorithm. This algorithm divides the planning horizon into equally sized blocks, and solves Lagrangian MISOCP subproblems. To the best of our knowledge, this is the first time MISOCP relaxations are utilized within a Lagrangian decomposition framework for the UC Problem with AC power flows. With the decomposition algorithm, we are able to solve larger instances within a time limit that we set to be one hour, with small optimality gaps.

The main contributions of this paper are as follows:

- We propose MISOCP-based solution algorithms for the UC Problem with AC power flows that exploit the recent advances in the convexification of the OPF Problem.
- We develop a novel MISOCP-based Lagrangian decomposition algorithm to solve the UC Problem with AC power flows.
- We provide a library of publicly available instances for the UC Problem with AC power flows to facilitate future research.

The rest of the paper is organized as follows: In Section II, we define the problems and formulations. In Section III, we introduce our proposed solution algorithms. In Section IV, we report the computational results. We conclude our paper in Section V with final remarks and comments on possible future work.

II. PROBLEM DEFINITIONS AND FORMULATIONS

A. Multi-period Optimal Power Flow

Consider a power network \( \mathcal{N} = (\mathcal{B}, \mathcal{L}) \) in which the set \( \mathcal{B} \) denotes the set of buses and \( \mathcal{L} \) denotes the set of transmission lines. Generators are attached to the buses, which are denoted by the set \( \mathcal{G} \subseteq \mathcal{B} \). With some abuse of notation, we refer to the transformers as transmission lines as well. However, their physical characteristics are correctly modeled in the formulation. Given the demand of each bus at each time period in a planning horizon \( \mathcal{T} \), the MOPF Problem aims to determine the amount of energy produced at each generator, and the economic dispatch of the active and reactive energy.

Note that the single-period OPF Problem is just a special case of the MOPF Problem with \( |\mathcal{T}| = 1 \) and without the ramping constraint. On the other hand, the MOPF Problem is a special case of the UC Problem for the given generator commitment statuses.

1) Formulation: The classical formulation of the OPF and MOPF Problems either uses the rectangular (or \( e - f \) ) formulation or the polar (or \( V - \theta \) ) formulation. We instead use the alternative formulation first proposed by [35] for the OPF Problem, which uses \( c - s - \theta \) variables as follows:

\[
\begin{align*}
    c_{i,t} := |V_{i,t}|^2 & \quad i \in \mathcal{B}, t \in \mathcal{T} \quad (1a) \\
    c_{ij,t} := |V_{i,t}| |V_{j,t}| \cos(\theta_{i,t} - \theta_{j,t}) & \quad (i,j) \in \mathcal{L}, t \in \mathcal{T} \quad (1b) \\
    s_{ij,t} := -|V_{i,t}| |V_{j,t}| \sin(\theta_{i,t} - \theta_{j,t}) & \quad (i,j) \in \mathcal{L}, t \in \mathcal{T} \quad (1c)
\end{align*}
\]

For these new variables, we obtain the following variable bounds:

\[
\begin{align*}
    V_{i,t}^2 & \leq c_{i,t} \leq \bar{V}_{i}^2 & \quad i \in \mathcal{B}, t \in \mathcal{T} \quad (2a) \\
    V_i V_j \cos(\bar{\theta}_{ij}) & \leq c_{ij,t} \leq V_i V_j & \quad (i,j) \in \mathcal{L}, t \in \mathcal{T} \quad (2b) \\
    V_i V_j \sin(\bar{\theta}_{ij}) & \leq s_{ij,t} \leq V_i V_j \sin(\bar{\theta}_{ij}) & \quad (i,j) \in \mathcal{L}, t \in \mathcal{T} \quad (2c)
\end{align*}
\]

We are now ready to provide the formulation of the MOPF Problem:

\[
\begin{align*}
    \min & \sum_{i \in \mathcal{B}} \sum_{t \in \mathcal{T}} C_i(p_{i,t}^q) \\
    \text{s.t.} & \quad \text{For each } i \in \mathcal{B}, t \in \mathcal{T} : \\
    & p_{i,t}^d - p_{i,t}^q = g_{ii} c_{i,t} + \sum_{j \in \delta(i)} (\overline{p}_{ij,t} + \overline{p}_{ij,t}) & \quad (3b) \\
    & q_{i,t}^d - q_{i,t}^q = -b_{ii} c_{i,t} + \sum_{j \in \delta(i)} (\overline{q}_{ij,t} + \overline{q}_{ij,t}) & \quad (3c) \\
    & q_{i,t}^q \leq q_{i,t}^d & \quad \forall i \in \mathcal{B} & \quad (3d) \\
    & p_{i,t}^q \leq p_{i,t}^d & \quad \forall i \in \mathcal{B} & \quad (3e) \\
    & -RD_i \leq p_{i,t}^d - p_{i,t}^q \leq RU_i & \quad \forall i \in \mathcal{B} & \quad (3f)
\end{align*}
\]

For each \( (i,j) \in \mathcal{L}, t \in \mathcal{T} : 

\[
\begin{align*}
    & \overline{p}_{ij,t} = G_{ij} f_{ij} c_{ij,t} + G_{ij} f_{ij} c_{ij,t} - B_{ij} f_{ij} s_{ij,t} \quad (3g) \\
    & \overline{q}_{ij,t} = G_{ij} f_{ij} c_{ij,t} + G_{ij} f_{ij} c_{ij,t} - B_{ij} f_{ij} s_{ij,t} \quad (3h) \\
    & \overline{q}_{ij,t} = -B_{ij} f_{ij} c_{ij,t} - B_{ij} f_{ij} c_{ij,t} - G_{ij} s_{ij,t} \quad (3i) \\
    & (\overline{p}_{ij,t})^2 + (\overline{q}_{ij,t})^2 \leq \mathcal{I}^2 \quad (3k) \\
    & (\overline{p}_{ij,t})^2 + (\overline{q}_{ij,t})^2 \leq \mathcal{I}^2 \quad (3l) \\
    & (c_{ij,t})^2 + (s_{ij,t})^2 = c_{ii,t} c_{jj,t} \quad (3m) \\
    & s_{ij,t} = \tan(\theta_{ij,t} - \theta_{i,t}) c_{ij,t} \quad (2a) - (2e).
\end{align*}
\]
The objective function (3a) is a quadratic function of the real power generation amount. Constraints (3b) and (3c) are real and reactive power balances at bus $i$, respectively. Constraints (3d) and (3e) are bounds for reactive and real power outputs of generator $i$, respectively. Constraint (3f) is the ramping constraint, which ensures that the active power generation amount does not change rapidly from one period to another. Notice that the ramping constraint requires the historical data regarding the real power dispatch variables. To model this aspect of the problem correctly, we use a parameter $t'$ in this constraint. If we have the historical data for $t = 0$, then $t'$ is simply set to $t - 1$. Since this information is not present in our instances, we assume that the demand pattern is cyclic and define

$$
t' = \begin{cases} 
    t - 1 & \text{if } t \neq 1 \\
    24 & \text{if } t = 1
\end{cases}
$$

Constraints (3g) and (3h) are real power flows from bus $i$ to $j$ and from bus $j$ to $i$, respectively. Constraints (3i) and (3j) are reactive power flows from bus $i$ to $j$ and from bus $j$ to $i$, respectively. Constraints (3k) and (3l) are power flow upper bounds for line $(i, j)$, for forward and backward flow, respectively. Constraints (3m) and (3n) are the consistency constraints for the relation between $c - s$ variables and original variables of the problem.

2) SOCP Relaxation: An SOCP relaxation of (3) is obtained by relaxing the constraint (3m) and omitting the constraint (3n).

$$\min \sum_{t \in T} \sum_{i \in G} C_i(p_{i,t}^g) \quad (4a)$$

s.t. $(c_{ij,t})^2 + (s_{ij,t})^2 \leq c_{ii,t}c_{jj,t} \quad (i,j) \in L, t \in T$

$$\quad (2a) - (2c), (3b) - (3d). \quad (4b)$$

This SOCP relaxation is utilized for constructing an MISOCO relaxation of the UC Problem with AC power flows in Section II-B2.

3) DC Approach: Since LP’s are efficiently solvable, linearized AC power flow equations are used with the assumption of a DC power grid. The DC approximation of the AC MOPF is formulated as follows:

$$\min \sum_{t \in T} \sum_{i \in G} C_i(p_{i,t}^g) \quad (5a)$$

s.t. For each $i \in B$, $t \in T$ :

$$p_{i,t}^g - p_{i,t}^d = \sum_{j \in b(i)} f_{ij,t} \quad (5b)$$

For each $(i,j) \in L$, $t \in T$ :

$$f_{ij,t} = - f_{ji,t} = B_{ij} (\theta_{i,t} - \theta_{j,t}) \quad (5c)$$

$$- S_{ij} \leq f_{ij,t} \leq S_{ij} \quad (3e) - (3f). \quad (5d)$$

In this formulation, constraint (5b) enforces flow conservation in each bus $i$. Constraint (5c) defines real power across the line $(i,j)$. Constraint (5d) enforces an upper bound on the power over line $(i,j)$. We will make use of this formulation to construct an MILP approximation of the UC Problem with AC power flows in Section II-B3.

B. Unit Commitment

The aim of the UC Problem is to determine the commitment schedule of generators over a time horizon given the energy demand of the power system in a way that minimizes system-wide energy generation cost, fixed and startup costs of generators while satisfying the operational requirements. Generally, the commitment schedule is obtained with respect to an approximate DC power flow in the UC level. In our study, we aim to solve the UC Problem while considering the AC power flows, therefore, the problem is challenging.

1) Formulation: We present our UC formulation with AC power flows below that utilizes the MOPF formulation from Section II-A1 with the addition of commitment, startup and shutdown status variables. Recall that since we do not have the generator commitment history at hand, we assume a cyclic demand pattern. To account for this modeling assumption, we define the following index sets for minimum uptime and downtime constraints:

$$T_{i,t}^{up} := \{ x : x = (t - MinU p_i + j) \text{mod} 24 + 1, \quad j \in \mathbb{Z}_0^+, 0 \leq j \leq MinU p_i - 1 \}, \forall g \in G, \forall t \in T$$

$$T_{i,t}^{dw} := \{ x : x = (t - MinD w_i + j) \text{mod} 24 + 1, \quad j \in \mathbb{Z}_0^+, 0 \leq j \leq MinD w_i - 1 \}, \forall g \in G, \forall t \in T.$$

Then, we are able to present the formulation as follows:

$$\min \sum_{t \in T} \sum_{i \in G} (f_{i,t}(u_{i,t}, v_{i,t}, w_{i,t}) + c_{i,t}(p_{i,t}^g)) \quad (6a)$$

s.t. For each $i \in G$, $t \in T$ :

$$u_{i,t'} - u_{i,t} = v_{i,t} - w_{i,t} \quad (6b)$$

$$v_{i,t} - u_{i,t} \leq 0 \quad (6c)$$

$$w_{i,t} + u_{i,t} \leq 1 \quad (6d)$$

$$\sum_{\tau \in T_{i,t}^{up}} v_{i,\tau} \leq u_{i,t} \quad (6e)$$

$$\sum_{\tau \in T_{i,t}^{dw}} w_{i,\tau} \leq 1 - u_{i,t} \quad (6f)$$

$$p_{i,t}^{min} u_{i,t} \leq p_{i,t}^g \leq p_{i,t}^{max} u_{i,t} \quad (6g)$$

$$q_{i,t}^{min} u_{i,t} \leq q_{i,t}^g \leq q_{i,t}^{max} u_{i,t} \quad (6h)$$

$$u_{i,t}, v_{i,t}, w_{i,t} \in \{0, 1\}$$

$$\quad (2a) - (2c), (3b) - (3c), (3f) - (3n). \quad (6i)$$

The first component of the objective function (6a) is a linear function of the binary variables: commitment status, startup status and shutdown status of the generators. The second component of the objective function is a function of active power generation of generators, and generally is a quadratic function. Constraints (6b), (6c) and (6d) are logical constraints for the
Algorithm 1: Base Algorithm.

**Result:** Lower and upper bounds for UC Problem with AC power flows

1: Solve the MISOCO relaxation (7) of Problem (6).
2: Obtain a candidate commitment schedule.
3: Solve Problem (3) with the commitment schedule obtained in the previous step.
4: Calculate the optimality gap.

Algorithm 2: Enhanced Algorithm.

**Result:** Lower and upper bounds for UC Problem with AC power flows

1: Compute a cycle basis.
2: For each edge, add the arctangent constraints as in Appendix B1.
3: Solve the continuous relaxation of MISOCP (7).
4: For \( i = 1 \) to \( I \):
   1. Solve the separation problem (10) in Appendix B for each cycle in the cycle basis in parallel.
   2. Add the cuts obtained from the separation problem as in Appendix B2 and resolve the continuous relaxation of MISOCP (7).
5: Solve the MISOCP problem (7) with the cuts obtained.
6: Obtain a candidate commitment schedule.
7: Solve Problem (3) with the commitment schedule obtained in the previous step.
8: Calculate the optimality gap.

III. PROPOSED SOLUTION ALGORITHMS

We propose three MISOCP-based algorithms for solving the UC Problem with AC power flow equations: i) base, ii) enhanced and iii) decomposition-based. Base and enhanced algorithms rely on solving the MISOCO relaxations of the UC Problem, and then solving an MOPF Problem to find feasible solutions. The decomposition algorithm is proposed for solving large instances and relies on using Lagrangian decomposition to solve the time-decomposed MISOCP relaxations of the UC Problem.

A. Base Algorithm

The base algorithm depends on the observation that if the generator commitment schedules are at hand, then the UC Problem reduces to an instance of the MOPF Problem. We utilize this fact and solve an MISOCO relaxation to find a candidate commitment schedule and a lower bound for the UC Problem with AC power flows. Then, we solve an MOPF Problem with the given commitment status to come up with a feasible solution to the UC Problem. The outline of our approach is given in Algorithm 1.

B. Enhanced Algorithm

In order to improve the lower bound obtained from Algorithm 1, we use two different methods taken from [16], which are arctangent envelopes and SDP cut separation. The first method utilizes an outer-approximation of the feasible region of the arctangent constraint by adding linear inequalities. The second method solves an SDP separation problem over some cycles of the network given a solution to the MISOCP relaxation, which is solved by MOSEK [36]. From the separation problem, we obtain linear inequalities and add them as a cutting plane to the model. Both of these methods have been originally derived for the AC OPF Problem in [16]. We apply these methods to the multiperiod setting and utilize them to solve the UC problem with AC power flows. We present the details of our approach in Algorithm 2.

C. Decomposition Algorithm

We develop a decomposition algorithm since solving the MISOCP problem becomes computationally expensive for large-scale instances. In order to obtain lower bounds faster, we propose dividing the planning horizon into blocks, which are solved separately. For the constraints that include time indices belonging to different blocks (subproblems), we utilize the Lagrangian decomposition method. For deciding on the Lagrangian multipliers, we first solve the continuous relaxation of the MISOCP problem (7) and retrieve the optimal dual values of these constraints. We set these dual values of the constraints as the corresponding Lagrangian multipliers and solve the subproblems for each block. In order to ensure that solution of each subproblem outputs a feasible commitment schedule, we solve an intermediate problem, which we call the restricted MISOCP. In this restricted MISOCP, we fix a generator to be turned on/off in the whole planning horizon, if a generator is turned on/off in each of the 24 time periods. Then, for the generators that are turned on for some periods but that are off for some periods,
we fix their turned on periods. The solution to this MISOPC becomes our candidate commitment schedule and we solve the MOPF Problem accordingly. We also apply the subgradient algorithm (see, e.g., [37]) to update the Lagrangian multipliers and try to improve the lower bound.

Before presenting the formulation, we define the following parameters for each generator $i \in G$, time index $t \in T$ and block $b$:

$$t^u = \begin{cases} \ t + 1, & \text{if } t \neq 24 \\ \ 1, & \text{if } t = 24 \end{cases}$$

$$t_b := \min(\mathcal{T}_b)$$

$$\mathcal{T}_b := \max(\mathcal{T}_b)$$

$$T_{i,t}^{\text{intu}} := T_{i,t}^{up} \cap (\mathcal{T} \setminus \mathcal{T}_b)$$

$$T_{i,t}^{\text{intd}} := T_{i,t}^{dw} \cap (\mathcal{T} \setminus \mathcal{T}_b).$$

$$T_{i,r1} := \{ t: T_{i,t}^{up} \cap \mathcal{T}_b \neq 0, T_{i,t}^{up} \subseteq \mathcal{T}_b \}$$

$$T_{i,r2} := \{ t: T_{i,t}^{intu} \neq \emptyset \}$$

$$T_{i,r3} := \{ t: T_{i,t}^{dw} \cap \mathcal{T}_b \neq 0, T_{i,t}^{dw} \subseteq \mathcal{T}_b \}$$

$$T_{i,r4} := \{ t: T_{i,t}^{intd} \neq \emptyset \}.$$  

In order to formulate the Lagrangian subproblems, we utilize the MISOPC relaxation (7) and obtain the subproblems for each block $b$ as follows:

$$\text{min} \sum_{t \in \mathcal{T}_b} \sum_{i \in G} \left( f_{i,t}(u_{i,t}, v_{i,t}, w_{i,t}) + c_{i,t}(p_{i,t}^g) \right) + \sum_{i \in G} \lambda_{i,t}^{\text{up}} \left( p_{i,t}^g - RU_i \right) + \sum_{i \in G} \lambda_{i,t}^{\text{up}} \left( -p_{i,t}^g \right)$$

$$+ \sum_{i \in G} \lambda_{i,t}^{\text{dw}} \left( p_{i,t}^g - RD_i \right) + \sum_{i \in G} \lambda_{i,t}^{\text{dw}} \left( -p_{i,t}^g \right)$$

$$+ \sum_{i \in G} \lambda_{i,t}^{\text{log}} \left( -u_{i,t} + v_{i,t} - w_{i,t} + \log(u_{i,t}) + \log(v_{i,t}) \right)$$

$$- \sum_{t \in T_{i,r1}} \sum_{i \in G} \lambda_{i,t}^{up} \left( \sum_{k \in T_{i,t}^{up}} (v_{i,k}) - u_{i,t} \right)$$

$$+ \sum_{t \in T_{i,r2}} \sum_{i \in G} \lambda_{i,t}^{up} \left( \sum_{k \in T_{i,t}^{up}} (v_{i,k}) \right)$$

$$+ \sum_{t \in T_{i,r3}} \sum_{i \in G} \lambda_{i,t}^{dw} \left( \sum_{k \in T_{i,t}^{dw}} (u_{i,k}) + u_{i,t} - 1 \right)$$

$$+ \sum_{t \in T_{i,r4}} \sum_{i \in G} \lambda_{i,t}^{up} \left( \sum_{k \in T_{i,t}^{up}} (w_{i,k}) \right)$$

s.t. $\sum_{\tau \in T_{i,t}^{up}} v_{i,\tau} \leq u_{i,t}$, $i \in G$, $t \in \mathcal{T}_b$ : $T_{i,t}^{up} \subseteq \mathcal{T}_b$.

**Algorithm 3: Decomposition Algorithm.**

**Result:** Lower and upper bounds for UC with AC power flows

$LB = 0, UB = \infty$

1. Compute a cycle basis.

2. For each edge, add the arc tangent constraints as in Appendix B1.

3. Solve the continuous relaxation of the MISOPC (7).

4. For $i = 1$ to $I$:

   1. Solve the separation problem (10) in Appendix B for each cycle in the cycle basis in parallel.

   2. Add the cuts obtained from the separation problem as in Appendix B2 and resolve the continuous relaxation of MISOPC (7).

5. Obtain the Lagrangian multipliers from the previous step.

6. For $i = 1$ to $J$:

   1. For each block $b$, solve the Lagrangian Relaxation Problem (9) and obtain $LB_i$.

   If $LB_i > LB$, $LB \leftarrow LB_i$.

   2. Obtain the commitment decisions for the original time horizon, solve the restricted MISOPC problem explained in Section III-C, and obtain $UB_i$.

   If $UB_i < UB$, $UB \leftarrow UB_i$.

3. Solve Problem (3) in order to find a feasible solution to the UC with AC power flows problem.

4. Update the Lagrangian Multiplier values with the subgradient algorithm.

7. Calculate the optimality gap.

$$\sum_{\tau \in T_{i,t}^{up}} w_{i,\tau} \leq 1 - u_{i,t} \quad i \in G, t \in \mathcal{T}_b : T_{i,t}^{up} \subseteq \mathcal{T}_b$$

$$u_{i,T_i} - u_{i,t} + v_{i,t} - w_{i,t} = 0 \quad i \in G, t \in \mathcal{T}_b \setminus \{ T_b \}$$

$$-RD_i \leq p_{i,t}^g - p_{i,T_i}^g \quad i \in G, t \in \mathcal{T}_b \setminus \{ T_b \}$$

$$p_{i,t}^g - p_{i,T_i}^g \leq RU_i \quad i \in G, t \in \mathcal{T}_b \setminus \{ T_b \}$$

$$\frac{1}{2} (2a) - (2c), (3b) - (3c), (3f) - (3l), (4b), (6g) - (6i).$$

Our decomposition method is outlined in Algorithm 3.

**IV. COMPUTATIONAL RESULTS**

**A. Instance Creation**

We face some difficulties in finding realistic problem instances for the UC Problem with AC power flows. Therefore, we decide to create our own problem instances based on NESTA AC OPF instances [38]. To be able to convert the AC OPF instance to an UC instance, we need the following parameters for each generator $i \in G$:

- ramp-up/down rate, minimum up/down time, startup cost and fixed cost. Also, for each bus $i \in B$, we need the demand for 24 periods. Due to lack of generator history, we assumed that the demand is cyclic. The details of the creation procedure are provided in Appendix A.

Instances
TABLE I

| Instance    | Method          | LBT(s) | UBT(s) | UB    | %Gap |
|-------------|-----------------|--------|--------|-------|------|
| 9wccc-SAD   | DC              | infas. |        |       |      |
| 9wccc-SAD   | MISOCP          | 0.15   | 1.15   | 1582.36 | 0.52 |
| 9wccc-SAD   | MISOCP++        | 4.58   | 1.06   | 1582.36 | 0.04 |
| 14iee-TYP   | DC              | 0.02   | -      | infas. | -    |
| 14iee-TYP   | MISOCP          | 2.49   | 2.05   | 227.13 | 0.03 |
| 14iee-TYP   | MISOCP++        | 13.64  | 2.24   | 227.13 | 0.00 |
| 57iee-SAD   | DC              | 0.13   | 7.26   | 642.30 | -    |
| 57iee-SAD   | MISOCP          | 8.66   | 7.17   | 545.44 | 0.77 |
| 57iee-SAD   | MISOCP++        | 47.13  | 7.15   | 545.44 | 0.84 |

created according to this procedure can be found at the following websites: https://sites.google.com/site/burakkocuk/research and https://github.com/deniztuncer/UC-with-ACOPF.

B. Computational Setting

The experiments are carried out on a desktop workstation with two 3.0 GHz processors and 256 GB of RAM. Lower bounds are obtained by solving MISOCP, utilizing Gurobi 9.5.0 [39]. Feasible solutions are obtained by IPOPT [40], a local solver, which provides upper bounds. We solve three types of instances which we denote by TYP, API and SAD and they refer to typical, congested and small angle difference condition instances, respectively. DC, MISOCP and MISOCP++ refer to the DC approach, base algorithm and enhanced algorithm, respectively. LBT and UBT refer to the time to find lower bound and upper bound, respectively. LB and UB refers to the lower bound and upper bound, respectively. ‘local inf’ denotes that IPOPT converged to a locally infeasible point. We calculate %Gap according to the following formula: $\frac{UB - LB}{UB} \times 100$. Inspired by [41], we also report the average violation of constraint (3n) over cycles in the column ‘Mismatch’ to quantify the relaxation quality from a different perspective. Given a solution obtained from the MISOCP relaxation as $(c_{ij}, s_{ij})$, we compute this measure as

$$\frac{1}{|C|} \sum_{C \in \mathcal{C}} \frac{1}{|T|} \sum_{t \in T} \left| \frac{1}{|C|} \sum_{(i,j) \in \mathcal{C}} \arctan\left(s_{ij,t}/c_{ij,t}\right) \right|,$$

where $C$ is a cycle basis of the network. Notice that this measure is averaged over the elements in the cycle basis and time periods. We set the relative optimality tolerance of Gurobi as 0.1% unless otherwise stated. For enhanced and decomposition algorithms, we set $I = J = 5$.

C. DC vs MISOCP Based Methods

We first run some preliminary experiments to compare the DC and MISOCP based methods. The computational results presented in Table I show that the DC approach generally finds a solution in a shorter time than the MISOCP based methods (with some abuse of terminology, we report the computational time of solving the DC based method under the LBT column).

We observe that the DC approach may yield unfavorable results compared to the MISOCP approach. Firstly, the DC UC Problem might be infeasible whereas our MISOCP-based approach gives AC-feasible solutions with small optimality gaps, as evident in 9wsc-s-SAD. Secondly, although the DC UC Problem is feasible, the candidate generator commitment schedule might not provide AC-feasible solutions, as evident in 14iee-TYP (in this case, our MISOCP-based approach gives AC-feasible solutions with small optimality gaps again). Finally, the candidate generator commitment schedule obtained by the DC UC approach may be suboptimal compared to the schedule obtained by our MISOCP-based approach, as evident in 57iee-s-SAD.

These three examples clearly show that the MISOCP-based solution approach is consistently more successful than the DC-based solution approach. Hence, we solely use MISOCP-based solution approaches in the remainder of the paper.

D. Computational Results for the Base and Enhanced Algorithms

In Table II, we present the computational results for the base and the enhanced algorithm.

The MISOCP method fails to produce a feasible commitment schedule in one of the congested instances and one of the small angle difference instances, whereas the MISOCP++ method is able to provide a feasible commitment schedule for each instance. Out of the instances that a feasible commitment schedule is obtained, MISOCP method and MISOCP++ method have 0.46% and 0.20% optimality gap in average, respectively. For TYP instances, the MISOCP++ outperforms MISOCP, by reducing average optimality gap from 0.34% to 0.06%. For API instances, the MISOCP method is able to solve seven out of eight instances with the average optimality gap of 0.47% whereas the MISOCP++ method yields a feasible solution to all eight instances with average optimality gap of 0.36%. Especially for the SAD instances, seven out of eight instances could be solved with MISOCP with an average optimality gap of 0.59%, whereas MISOCP++ yields feasible solutions to all eight instances with an average optimality gap of 0.25%.

The average mismatch of MISOCP and MISOCP++ methods are 0.0045 and 0.0005, respectively. Therefore, we conclude that the relaxation solutions obtained from the latter method have smaller mismatch with respect to the omitted constraint (3n) over cycles. This is another indication that the MISOCP++ method is more suitable. We would like to re-iterate that although the mismatch might be positive for the solutions obtained from the relaxations, we obtain feasible solutions for the nonconvex UC Problem (6) via the local solver IPOPT, which should have zero mismatch up to the feasibility tolerance.

The average time to obtain a lower bound for the problem using MISOCP and MISOCP++ are 6.9 and 24.1 seconds, respectively. An increase in the time to obtain a lower bound is expected since we are solving SDP separation problems in addition to the SOCP relaxations in the latter method. However, the additional computational effort is justified since we obtain stronger lower bounds and, hence, smaller optimality gaps. We also experiment with different cost combinations than reported in the paper and obtain similarly successful results [42].

We see that all the problem instances were solved with the MISOCP++ method to a less than 1.3% optimality gap.
and within a reasonable timespan, considering the challenging nature of the problem. However, for larger instances, finding a lower bound is time consuming. Therefore, we came up with a decomposition method in order to solve larger instances.

### E. Decomposition Method

In Table III, we present the computational results for the decomposition algorithm. Lagrangian subproblems are solved to 1% optimality tolerances. We note that we terminate the subgradient iterations once a feasible solution of the UC Problem with 1% proven optimality gap is obtained.

For the case 89peg-API, we are able to find a solution with 0.05% optimality gap in about two minutes. For the case 89peg-API, we are able to improve the lower bound with 0.05% optimality gap in about two minutes. For the case 89peg-API, we are able to improve the lower bound with 0.05% optimality gap in about two minutes.

### TABLE II

RESULTS FOR MISOCP AND MISOCP++

| Instance | Type  | LBT (s) | UBT (s) | LB Mismatch | UB | %Gap | LBT (s) | UBT (s) | LB Mismatch | UB | %Gap |
|----------|-------|---------|---------|-------------|----|------|---------|---------|-------------|----|------|
| 6ww      | TYP   | 0.14    | 0.97    | 4417.38     | 0.004 | 0.02 | 9.13    | 1.18    | 4417.38     | 0.000 | 0.00 |
| 9wsc     | TYP   | 2.49    | 2.05    | 227.13      | 0.004 | 0.03 | 13.64   | 2.24    | 227.13      | 0.000 | 0.00 |
| 14ieee   | TYP   | 0.74    | 4.28    | 15516.46    | 0.003 | 0.00 | 24.74   | 4.35    | 15516.46    | 0.000 | 0.00 |
| 24ieee   | TYP   | 0.68    | 3.04    | 1832.06     | 0.002 | 0.00 | 18.71   | 3.10    | 1832.06     | 0.000 | 0.00 |
| 30as     | TYP   | 8.62    | 3.70    | 235.44      | 0.004 | 0.21 | 21.74   | 4.52    | 235.44      | 0.000 | 0.00 |
| 30as     | TYP   | 26.01   | 5.67    | 28336.21    | 0.001 | 0.51 | 71.60   | 5.41    | 28336.21    | 0.003 | 0.03 |
| 39ppi    | TYP   | 16.22   | 6.63    | 542.08      | 0.004 | 0.07 | 45.79   | 6.71    | 542.08      | 0.002 | 0.02 |

### TABLE III

RESULTS FOR THE DECOMPOSITION ALGORITHM

| Instance | Type  | $b$ | LBT (s) | UBT (s) | LB | UB | %Gap | Total Time (s) | LB | UB | %Gap |
|----------|-------|----|---------|---------|----|----|------|----------------|----|----|------|
| 57iieee  | TYP   | 4  | 39.13   | 7.82    | 541.94 | 542.08 | 0.03 | 2503.32 | 1546.09 | 1722.95 | 10.26 |
| 57iieee  | TYP   | 4  | 36.47   | 27.45   | 1537.58 | 2163.77 | 40.73 | 2763.36 | 1559.37 | 2037.54 | 23.46 |
| 89peg    | TYP   | 4  | 26.64   | 103.77  | 2343.29 | 2344.56 | 0.05 | 1344.71 | 4946.81 | 54954.02 | 9.99 |
| 89peg    | TYP   | 4  | 26.64   | 104.47  | 2343.29 | 2344.56 | 0.05 | 1344.71 | 4946.81 | 54954.02 | 9.99 |
| 89peg    | API   | 4  | 35.14   | 13.52   | 653.12 | 652.30 | 0.03 | 22717.99 | 49941.70 | 52984.97 | 5.74 |
| 89peg    | API   | 6  | 37.12   | 13.81   | 653.12 | 652.30 | 0.03 | 22717.99 | 49941.70 | 52984.97 | 5.74 |
| 500goc   | TYP   | 4  | 288.27  | 157.33  | 664327.99 | 664777.50 | 0.07 | 223980. | 1828.55 | 2292.07 | 20.22 |
| 500goc   | TYP   | 6  | 298.11  | 160.45  | 664327.99 | 664777.50 | 0.07 | 223980. | 1828.55 | 2292.07 | 20.22 |
| 500goc   | API   | 4  | 32.33   | 9.17    | 543.64 | 545.44 | 0.33 | 22717.99 | 49941.70 | 52984.97 | 5.74 |
| 500goc   | API   | 6  | 32.81   | 9.32    | 543.93 | 545.44 | 0.28 | 22717.99 | 49941.70 | 52984.97 | 5.74 |
| 89peg    | SAD   | 4  | 25.48   | 80.43   | 53657.02 | 55683.04 | 0.05 | 223980. | 1828.55 | 2292.07 | 20.22 |
| 89peg    | SAD   | 6  | 27.91   | 82.01   | 53657.84 | 55683.04 | 0.05 | 223980. | 1828.55 | 2292.07 | 20.22 |
| 89peg    | SAD   | 6  | 499.63  | 58.96   | 1816.61 | 2278.83 | 25.44 | 223980. | 1828.55 | 2292.07 | 20.22 |
| 118iieee | SAD   | 4  | 169.32  | 844.74  | 626578.13 | 627438.64 | 0.14 | 223980. | 1828.55 | 2292.07 | 20.22 |
| 118iieee | SAD   | 6  | 187.83  | 843.26  | 626577.23 | 627438.64 | 0.14 | 223980. | 1828.55 | 2292.07 | 20.22 |
the subgradient iterations, therefore improving the optimality gap from 6.26% to 5.74%.

For the 118-bus instances, which are the most challenging instances we consider, our decomposition method manages to output a feasible solution. For these instances, the base and enhanced algorithms fail to produce a feasible solution with the commitment schedule we obtain within an hour. For the case 118ieee-TYP, the subgradient iterations lead to an improved feasible solution and reduce the optimality gap from 40.73% to 10.26%. For the case 118ieee-SAD, we have a relatively large optimality gap, due to the operating conditions and density of the network. We would like to note that such larger optimality gaps are not rare in the OPF literature [43].

For the case 500goc, which is an instance from the Grid Optimization Challenge [44], the decomposition method outputs a feasible solution with less than 0.01% optimality gap in less than 15 minutes under typical operating conditions. For cases 500goc-API and 500goc-SAD, we achieve respective optimality gaps of 0.07% and 0.14%.

In 7 out of 11 instances we report in Table III, setting \( b = 4 \) and \( b = 6 \) yield similar results. For two of the remaining instances (57ieee-SAD and 118ieee-SAD), setting \( b = 6 \) yields slightly better optimality gaps compared to setting \( b = 4 \). For the remaining two instances (118ieee-TYP and 89peg-API), setting \( b = 4 \) yields significantly better optimality gaps compared to setting \( b = 4 \). Overall, our computational results suggest that dividing the planning horizon into a smaller number of blocks may yield better results as we relax less constraints. In addition, applying the decomposition method to the 57-bus instances results in solving the problem in less time, while achieving similarly successful objective values compared to the MIOSCP++ method.

Finally, we note that we face numerical difficulties for the 118ieee-API instance and are not able to obtain a feasible solution. Also, for 89peg instances, we only add arc tangent envelopes but not SDP inequalities, due to numerical difficulties.

### V. CONCLUSION AND FUTURE WORK

In this paper, we studied the UC Problem with AC power flow equations. We developed three solution methods that utilize the MIOSCP relaxation of this challenging problem. Since there was a lack of publicly available problem instances, we constructed realistic UC Problem instances with AC power flows based on publicly available AC OPF instances. We empirically showed that the classical DC-based approach might yield unfavorable generator commitment schedules whereas our MIOSCP-based solution approaches consistently provided provably high-quality feasible solutions over these instances. In particular, we were able to provide a feasible solution within 1.3% optimality gap over instances with up to 57-bus and 24-hour planning horizon in at most two minutes of computational time. To solve even larger instances, we developed a decomposition method based on the Lagrangian relaxation approach. The decomposition method enabled us to solve instances with up to 500 buses.

There are several promising future research directions. From a modeling point of view, we can explore a more realistic formulation that has a 15-minute time resolution and considers the uncertainty of power units such as hydro units. We have already conducted preliminary experiments for the former case with one-hour commitment and 15-minute dispatch intervals for a 24-hour planning horizon. The results presented in Appendix C are promising but we concur that a more thorough analysis is needed to obtain more satisfactory results. From an algorithmic point of view, we can improve the decomposition method to solve larger problem instances with more detailed formulations. We anticipate that both temporal and spatial decompositions would be needed in these cases.

### APPENDIX A

#### INSTANCE CREATION DETAILS

For each bus, we utilize the real demand profiles taken from [45], [46] and [33] and randomly assign these profiles. We assume that peak demand of the demand profile is equal to the demand of the AC OPF instance and normalize the demand accordingly. For the reactive demand, we utilize the reactive demand from [33]. The demand profiles are given in Table IV.

For the determination of UC-specific parameters for each generator, we randomly assign each generator a type, and then calculate the parameters based on their types. The calculation of the parameters are described as follows: For each generator \( i \in \mathcal{G} \), we assign three types, Type 1, 2 and 3.

- If generator \( i \) is of Type 1, then \( RU_i = RD_i = \max \left\{ p_i^{\min}, \frac{p_i^{\max}}{2} \right\}, MinUp_i = MinDw_i = 2 \).
- If generator \( i \) is of Type 2, then \( RU_i = RD_i = \max \left\{ p_i^{\min}, \frac{p_i^{\max}}{4} \right\}, MinUp_i = MinDw_i = 3 \).
- If generator \( i \) is of Type 3, then \( RU_i = RD_i = \max \left\{ p_i^{\min}, \frac{p_i^{\max}}{5} \right\}, MinUp_i = MinDw_i = 4 \).

#### TABLE IV

| Time period | Real Profile 1 | Real Profile 2 | Real Profile 3 | Reactive Profile |
|-------------|----------------|----------------|----------------|------------------|
| 1           | 0.68           | 0.57           | 0.67           | 0.68             |
| 2           | 0.64           | 0.64           | 0.63           | 0.65             |
| 3           | 0.61           | 0.68           | 0.60           | 0.62             |
| 4           | 0.60           | 0.71           | 0.59           | 0.60             |
| 5           | 0.60           | 0.75           | 0.59           | 0.61             |
| 6           | 0.62           | 0.78           | 0.60           | 0.63             |
| 7           | 0.67           | 0.82           | 0.74           | 0.68             |
| 8           | 0.74           | 0.85           | 0.86           | 0.69             |
| 9           | 0.80           | 0.88           | 0.95           | 0.73             |
| 10          | 0.84           | 0.92           | 0.96           | 0.81             |
| 11          | 0.89           | 0.97           | 0.96           | 0.89             |
| 12          | 1.00           | 0.90           | 0.95           | 0.92             |
| 13          | 0.93           | 0.92           | 0.95           | 0.95             |
| 14          | 0.95           | 0.88           | 0.95           | 0.95             |
| 15          | 0.97           | 0.85           | 0.93           | 0.97             |
| 16          | 0.99           | 0.78           | 0.94           | 1.00             |
| 17          | 1.00           | 0.71           | 0.99           | 1.00             |
| 18          | 0.96           | 0.78           | 1.00           | 0.96             |
| 19          | 0.96           | 0.85           | 1.00           | 0.96             |
| 20          | 0.92           | 0.92           | 0.96           | 0.93             |
| 21          | 0.92           | 0.85           | 0.91           | 0.93             |
| 22          | 0.88           | 0.78           | 0.83           | 0.91             |
| 23          | 0.78           | 0.71           | 0.73           | 0.77             |
| 24          | 0.76           | 0.64           | 0.63           | 0.76             |
We choose the following cost combinations for the fixed and startup costs of the generators, respectively: $F_i = 5 L_{t_i}$ and $SU_{p_i} = 100 L_{t_i}$, where $L_{t_i}$ is the linear cost of power generation. The shutdown cost is selected as zero in accordance with the literature.

**APPENDIX B**

**DETAILS OF THE ENHANCED ALGORITHM**

Base algorithm is able to quickly produce solutions for the UC Problem with small optimality gaps. For obtaining even smaller optimality gaps, it may be useful to improve the lower bound obtained by the MISOCOP problem (7). To improve lower bounds, we use two methods originally derived for the OPF Problem [16] and adapt it to the MOPF Problem. We briefly explain these two methods that are used in the Enhanced Algorithm.

**A Arctangent Envelopes**

Constraint (3m) includes an arctangent function that is non-convex. For convexifying this constraint, four linear envelopes can be utilized. We will now explain how to find these envelopes.

For each line $(i, j)$, let $\theta_{i,j} = \theta_i - \theta_j$. For simplicity, if $i$ and $j$ indices are dropped, the equation can be written as $\theta = \arctan(s/c)$, with $(c, s) \in [c, \pi] \times [s, \pi]$.

Assuming $c > 0$, the four corners of the box constraints are defined in the $(c, s, \theta)$ space as:

- $z^1 = (c, \pi, \arctan(\pi/c))$,
- $z^2 = (\pi, \pi, \arctan(\pi/c))$,
- $z^3 = (\pi, s, \arctan(s/\pi))$,
- $z^4 = (c, s, \arctan(s/\pi))$.

Then, the following can be computed:

- $\theta = \gamma_1 + \alpha_1 c + \beta_1 s$, the plane that passes through $\{z^1, z^2, z^3\}$,
- $\theta = \gamma_2 + \alpha_2 c + \beta_2 s$, the plane that passes through $\{z^1, z^4, z^3\}$,
- $\theta = \gamma_3 + \alpha_3 c + \beta_3 s$, the plane that passes through $\{z^1, z^2, z^4\}$,
- $\theta = \gamma_4 + \alpha_4 c + \beta_4 s$, the plane that passes through $\{z^2, z^3, z^4\}$.

We can find the valid inequalities that approximate the upper envelope of the arctangent constraint by following the steps below:

- Find $\Delta \gamma_k$ by solving the optimization problem:
  $\Delta \gamma_k = \max\{\arctan(s/c) - (\gamma_k + \alpha_k c + \beta_k s) : c \in [c, \pi], s \in [s, \pi]\}$, for $k = 1, 2$,
- Let $\gamma_k' = \gamma_k + \Delta \gamma_k$,
- Add constraint $\gamma_k' + \alpha_k c + \beta_k s \geq \arctan(s/c)$, for $k = 1, 2$.

For the valid inequalities that approximate the lower envelope of the arctangent constraint, we can follow a similar procedure:

- Find $\Delta \gamma_k$ by solving the optimization problem:
  $\Delta \gamma_k = \max\{\gamma_k + \alpha_k c + \beta_k s - \arctan(s/c) : c \in [c, \pi], s \in [s, \pi]\}$, for $k = 3, 4$,
- Let $\gamma_k' = \gamma_k - \Delta \gamma_k$,
- Add constraint $\gamma_k' + \alpha_k c + \beta_k s \leq \arctan(s/c)$, for $k = 3, 4$.

For solving the optimization problems mentioned above, the Karush-Kuhn-Tucker (KKT) points are enumerated. By evaluating the objective function value at each KKT point, the point that yields the maximum objective function value is used.

**B SDP Separation**

In the literature, the SDP relaxation is utilized for solving the OPF Problem. Since solving the full SDP relaxation is computationally expensive, [16] suggests obtaining cutting planes that are generated from a cycle-based SDP relaxation. To start with, let us consider the following system of equations that are used in the full SDP relaxation:

$$
c_{ij} = W_{ij} + W_{ij}^r \quad (i, j) \in \mathcal{L}
$$

$$
s_{ij} = W_{ij} - W_{ji}^r \quad (i, j) \in \mathcal{L}
$$

$$
c_{ii} = W_{ii} + W_{ii}^r \quad i \in \mathcal{B}
$$

$$
W \succeq 0.
$$

If the condition rank($W$) = 1 is satisfied, then $W$ can be used to obtain a feasible for the UC Problem. However, since solving the full SDP relaxation is computationally expensive, we instead solve SDP relaxations for the cycles in the cycle basis of the network. For a given cycle $C$, we consider the matrix $\tilde{W} \in \mathbb{R}^{2|C| \times 2|C|}$, which is a submatrix of $W$, and consider the following set, where $z = (c, s)$ denotes the variables of the cycle $C$:

$$
S := \{ z \in \mathbb{R}^{2|C|} : \exists \tilde{W} \in \mathbb{R}^{2|C| \times 2|C|} : -z_l + A_l \bullet \tilde{W} = 0 \forall l \in L, \tilde{W} \succeq 0 \}.
$$

Here, the constraints $z_l = A_l \bullet \tilde{W}$ are taken from the full SDP relaxation that correspond to the cycle $C$.

Given a feasible solution $z^* = (c^*, s^*)$ of problem (7), the separation problem for the set $S$ is formulated as follows (here, $e$ is a vector with all entries equal to 1):

$$
v^* := \min \left\{ -\alpha^T z^* : \sum_{l \in L} \lambda_l A_l \succeq 0, \alpha \succeq 0, -e \leq \alpha \leq e \right\}.
$$

If the optimal value of this SDP, denoted by $v^*$, is strictly negative, then the inequality $\alpha^T z \leq 0$ can be added to the UC formulation as a cutting plane, where $\alpha^*$ is an optimal solution of the SDP separation problem (10).

**APPENDIX C**

**EXTENSION TO 96 TIME PERIODS**

In this section, we present the computational results for the UC problem with AC power flow equations in which the commitment decisions are kept hourly while the dispatch decisions are made for every 15-minutes within a 24-hour planning horizon. Hence, we can treat this case as an extension with 96 time periods. Such an extension is reasonable since the OPF Problem is typically solved multiple times in each hour of the day whereas commitment decisions are typically made for an hour.

We adjust some parameters to extend the problem into 96 periods. First, we assume that the change in the demand of
each bus is linear from one hour to another, therefore, we fill in the missing demand parameters according to a simple linear interpolation. Second, we divide the ramp up and ramp down rates of the generators by four to obtain the rates for 15-minutes.

We repeat the experiments in Table II under the described setting for small and medium size instances, and report the results in Table V. We first observe that the upper bounds are almost identical with those obtained with the 24-period instances, which is an expected outcome due to the way the new instances are created. We also report that the average running time of MISOCP and MISOCP++ methods increases from 10.49 seconds to 71.81 seconds, and from 27.83 seconds to 168.91 seconds, respectively. This increase is expected due to the increased number of AC power flow constraints. We note that the maximum time to obtain a solution in the MISOCP++ method is about 15 minutes, which is quite reasonable. We obtain similar optimality gaps compared to the 24-period setting, except for the instances 9wscc and 39epri under the typical and small angle difference conditions. For these instances, the feasible solutions are very similar to the 24-period setting, but since the lower bounds are weak, their optimality gaps are large. We think that the reason could be the selection of the ramp rates.

We also run the decomposition algorithm for the larger instances and report the results in Table VI similar to Table III for $b = 6$ and $b = 8$. We perform subgradient iterations 5 times if we do not exceed the time limit of one hour. If we have already reached the time limit, we do not carry out more iterations. The results obtained are somehow promising. Except for the

| Instance | Type | $b$ | LBT (s) | UBT (s) | LB | UB | %Gap | Total Time (s) | Solution before applying subgradient algorithm | Solution after at most 5 iterations of subgradient algorithm |
|----------|------|-----|---------|--------|----|----|------|---------------|-----------------------------------------------|-----------------------------------------------------|
| 57iwee   | TYP  | 6   | 186.13  | 46.49  | 541.94 | 542.06 | 0.02  | 2710.69       | 619351.97 | 626768.83 | 1.18 |
| 89preg   | TYP  | 6   | 92.45   | 637.88 | 2343.27 | 2344.51 | 0.05  | 2709.32       | 619349.79 | 626768.83 | 1.18 |
| 500goc   | TYP  | 8   | 398.94  | 1054.30 | 619351.96 | 626768.83 | 1.18  | 4343.63       | 49361.45 | 54857.79 | 10.02 |
| 57iwee   | API  | 6   | 251.49  | 38.96  | 631.89 | 632.29 | 0.07  | 4631.81       | 48999.04 | 54857.79 | 10.68 |
| 89preg   | API  | 6   | 978.34  | 1450.19 | 65459.81 | 664767.96 | 1.53  | 4128.18       | 654589.48 | 664767.96 | 1.53 |
| 500goc   | API  | 8   | 1006.30 | 1459.43 | 654588.48 | 664767.96 | 1.53  | 4196.08       | 654594.01 | 664767.96 | 1.53 |

| 57iwee   | SAD  | 6   | 450.95  | 36.87  | 527.41 | 545.42 | 3.30  | 1931.52       | 527.30  | 545.42 | 3.29 |
| 89preg   | SAD  | 6   | 216.48  | 337.69 | 53561.84 | 55233.72 | 3.03  | 2754.09       | 53643.11 | 55233.72 | 2.88 |
| 500goc   | SAD  | 8   | 632.46  | 1694.97 | 619127.87 | 627391.26 | 1.32  | 4367.55       | 619127.92 | 627391.26 | 1.32 |
ACKNOWLEDGMENT

The authors would like to thank The Scientific and Technological Research Council of Turkey (TÜBİTAK) for supporting this study with Project number 119M855.

REFERENCES

[1] S. Frank, I. Steponavič̆, and S. Rebennack, “Optimal power flow: A bibliographic survey I,” Energy Syst., vol. 3, no. 3, pp. 221–258, 2012.
[2] A. M. Sasson, F. Vilaroa, and F. Aboytes, “Optimal load flow solution using the hessian matrix,” IEEE Trans. Power Appar. Syst., vol. PAS-92, no. 1, pp. 31–41, Jan. 1973.
[3] G. da Costa, “Optimal reactive dispatch through primal-dual method,” IEEE Trans. Power Syst., vol. 12, no. 2, pp. 669–674, May 1997.
[4] G. Tognola and R. Bacher, “Unlimited point algorithm for OPF problems,” IEEE Trans. Power Syst., vol. 14, no. 3, pp. 1046–1054, Aug. 1999.
[5] W. Bulpsh, A. Groteh, K. McKinnon, and P. Trodden, “Local solutions of the optimal power flow problem,” IEEE Trans. Power Syst., vol. 28, no. 4, pp. 4780–4878, Nov. 2013.
[6] B. Kocuk, S. S. Dey, and X. A. Sun, “Inexactness of SDP relaxation and valid inequalities for optimal power flow,” IEEE Trans. Power Syst., vol. 31, no. 1, pp. 642–651, Jan. 2016.
[7] C. Coffrin and P. Van Hentenryck, “A linear-programming approximation of ac power flows,” INFORMS J. Comput., vol. 26, no. 4, pp. 718–734, 2012.
[8] B. Stott, J. Jardim, and O. Alsa, “DC power flow revisited,” IEEE Trans. Power Syst., vol. 24, no. 3, pp. 1290–1300, Aug. 2009.
[9] D. Bienstock and G. Munoz, “On linear relaxations of OPF problems,” 2014, arXiv:1411.1120.
[10] R. E. Griffith and R. A. Stewart, “A nonlinear programming technique for the optimization of continuous processing systems,” Manage. Sci., vol. 7, no. 4, pp. 379–392, 1961.
[11] X. Bai and H. Wei, “Semi-definite programming-based method for security-constrained unit commitment with operational and optimal power flow constraints,” Gener. Transmiss. Distribution IET, vol. 3, no. 3, pp. 182–197, 2009.
[12] X. Bai, H. Weiha, K. Fujisawa, and Y. Wang, “Semidefinite programming for optimal power flow problems,” Int. J. Elect. Power Energy Syst., vol. 30, no. 7, pp. 383–392, 2008.
[13] J. Lavaei and S. H. Low, “Zero duality gap in optimal power flow problem,” IEEE Trans. Power Syst., vol. 27, no. 1, pp. 92–107, Feb. 2012.
[14] R. Madani, M. Ashraphijuo, and J. Lavaei, “Promises of conic relaxation for contingency-constrained optimal power flow problem,” IEEE Trans. Power Syst., vol. 31, no. 2, pp. 1297–1307, Mar. 2016.
[15] R. Jabr, “Radial distribution load flow using conic programming,” IEEE Trans. Power Syst., vol. 21, no. 3, pp. 1458–1459, Aug. 2006.
[16] B. Kocuk, S. Dey, and X. Sun, “Strong socp relaxations for the optimal power flow problem,” Operations Res., vol. 64, pp. 1177–1196, 2016.
[17] A. Gurobi Optimization, “Gurobi optimizer reference manual,” 2021. [Online]. Available: https://www.gurobi.com
[18] A. Wächter and L. Biegler, “On the implementation of an interior-point algorithm for large-scale nonlinear optimization,” Math. Programming., vol. 103, pp. 41–63, 2005.
[45] S. Kazarlis, A. Bakirtzis, and V. Petridis, “A genetic algorithm solution to the unit commitment problem,” *IEEE Trans. Power Syst.*, vol. 11, no. 1, pp. 83–92, Feb. 1996.

[46] C. Grigg et al., “The IEEE reliability test system-1996. A report prepared by the reliability test system task force of the application of probability methods subcommittee,” *IEEE Trans. Power Syst.*, vol. 14, no. 3, pp. 1010–1020, Aug. 1999.

Deniz Tuncer received the B.S. degree in industrial engineering from Sabancı University, Istanbul, Turkey, with a minor degree in mathematics, and the M.S. degree in industrial engineering from Sabancı University, where he is currently working toward the Doctoral degree in industrial engineering.

Burak Kocuk received the B.S. degree in industrial engineering and mathematics, the M.S. degree in industrial engineering from Boğaziçi University, Istanbul, Turkey, and the Ph.D. degree in operations research from the School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA. He is currently an Associate Professor with the Industrial Engineering Program, Sabancı University, Istanbul, Turkey. Before joining Sabancı University, he was a Postdoctoral Fellow with the Tepper School of Business, Carnegie Mellon University, Pittsburgh, PA, USA. His research interests include mixed-integer nonlinear programming and stochastic optimization problems, from both theoretical and methodological aspects.