

A Distributed Mixed-Integer Framework to Stochastic Optimal Microgrid Control

Andrea Camisa¹, Member, IEEE, and Giuseppe Notarstefano¹, Member, IEEE

Abstract—This article deals with distributed control of microgrids composed of storages, generators, renewable energy sources, and critical and controllable loads. We consider a stochastic formulation of the optimal control problem associated with the microgrid that appropriately takes into account the unpredictable nature of the power generated by renewables. The resulting problem is a mixed-integer linear program and is NP-hard and nonconvex. Moreover, the peculiarity of the considered framework is that no central unit can be used to perform the optimization, but rather the units must cooperate with each other by means of neighboring communication. To solve the problem, we resort to a distributed methodology based on a primal decomposition approach. The resulting algorithm is able to compute high-quality feasible solutions to a two-stage stochastic optimization problem, for which we also provide a theoretical upper bound on the constraint violation. Finally, a Monte Carlo numerical computation on a scenario with a large number of devices shows the efficacy of the proposed distributed control approach. The numerical experiments are performed on realistic scenarios obtained from Generative Adversarial Networks (GANs) trained on an open-source historical dataset of the EU.

Index Terms—Distributed optimization, mixed-integer linear programming (MILP), stochastic microgrid control.

I. INTRODUCTION

IN THE last decade, the use of renewable energy sources is soaring and is creating new challenges in the field of microgrid control. These important structural changes in the power grid call for novel approaches that must appropriately take into account the stochastic nature of the energy produced by renewables. To this end, optimization-based control techniques are increasingly used. However, they typically employ centralized approaches that require the collection of the problem data at each node, which may lead to a single point of failure. Distributed optimization approaches are a promising alternative that allows for the solution of optimization problems with spatially distributed data while preserving the locality of the data and even resilience of the network in the

case of failures [1]–[3]. We first review optimal control techniques, and then, we recall approaches based on mixed-integer programming and, finally, move to distributed approaches. Optimal control techniques allow for shaping input trajectories that take into account energy consumption/production costs and user comfort. In recent times, they are increasingly achieved with moving horizon techniques, such as Model Predictive Control (MPC), since it flexibly allows one to tackle several challenges (see [4]–[6]). Stochastic optimization-based approaches are also being developed. In [7], a stochastic optimization method for energy and reserve scheduling with renewable energy sources and demand-side participation is considered. The work [8] studies a stochastic unit commitment and economic dispatch problem with renewables and incorporates the battery operating cost. Another prominent approach is mixed-integer linear programming (MILP), which is gathering significant attention due to its ability to model logical statements that often occur within microgrids. In [9], an MILP optimal control approach of residential microgrid is proposed. In [10], a mixed-integer nonlinear programming formulation is considered with experimental validation for islanded-mode microgrids. In [11], an MILP is formulated to achieve optimal load shifting in microgrids. The MPC and the MILP approaches have been combined in [12], which proposes a receding horizon implementation of the MILP approach on an experimental testbed. A stochastic version of this work is considered in [13], which further takes into account renewable energy sources and aims at an environmental/economical operation of microgrids. While these works take into account more and more aspects of microgrids, they are all based on centralized optimization techniques that require one of the nodes to be chosen as master, thus introducing scalability and privacy issues. As energy networks are intrinsically distributed, there is often the need to devise distributed approaches that exploit the graph structure. The recent survey [14] provides an overview of distributed control methods for microgrids. The work [1] reviews distributed methods for optimal power flow problems, while [15] surveys distributed control approaches for autonomous power grids. In [16], a distributed approach to optimal reactive power compensation is proposed. Causevic *et al.* [17] and Belluschi *et al.* [18] propose distributed algorithms for optimal energy building management, while Cavarero *et al.* [19] investigate a distributed feedback control law to minimize power generation cost in prosumer-based networks. However, none of the mentioned works formulates a comprehensive stochastic scheduling

Manuscript received 15 June 2021; revised 10 February 2022; accepted 18 April 2022. Date of publication 24 May 2022; date of current version 28 December 2022. This work was supported by the European Research Council (ERC) through the European Union’s Horizon 2020 Research and Innovation Programme under Grant Agreement 638992-OPT4SMART. Recommended by Associate Editor T. Yang. (Corresponding author: Andrea Camisa.)

The authors are with the Department of Electrical, Electronic and Information Engineering, University of Bologna, 40136 Bologna, Italy (e-mail: a.camisa@unibo.it; giuseppe.notarstefano@unibo.it).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TCST.2022.3174968>.

Digital Object Identifier 10.1109/TCST.2022.3174968

problem involving the demand-side in a distributed way. Novel distributed methods relying on MILPs can take advantage of the latest progress in distributed optimization methods. MILPs are nonconvex and NP-hard; therefore, large-scale instances can be solved within acceptable time windows only suboptimally. In this regard, the recent works [20], [21] propose distributed algorithms to compute feasible solutions of MILPs over networks. Specifically, the work [20] is based on a dual decomposition approach with an iterative restriction mechanism, while Camisa *et al.* [21] consider a primal decomposition scheme with a fixed restriction on the coupling constraints. However, both approaches require bounded constraint sets and may be severely penalized by the restriction of the coupling constraints.

The contributions of this article are given as follows. We consider a distributed stochastic microgrid control problem consisting of several interconnected power units, namely, generators, renewable energy sources, storages, and loads. We begin by recalling the microgrid model. We then show that the optimal control problem can be recast as a distributed MILP. We apply a two-stage stochastic programming approach to the distributed MILP and show that also this problem can be cast as a distributed MILP. The considered problems are both large-scale (i.e., with a large number of optimization variables) and mixed-integer (i.e., with some of the variables constrained to be an integer). Furthermore, we are considering a scenario where the problem has to be solved with a distributed algorithm, that is, with a peer-to-peer protocol without a central coordinator. From an algorithmic point of view, these three assumptions make the problem very challenging. We propose to tackle this distributed MILP using an approach inspired by recent approaches proposed in the literature, which are suitably modified to deal with the stochastic scenario. In order to adapt state-of-the-art distributed mixed-integer schemes for the control of microgrids, it will be necessary to extend the scope of such approaches to deal with unbounded constraint sets arising from the two-stage stochastic optimization approach. The proposed algorithm provides a feasible solution to the two-stage stochastic problem at each iteration while preserving sensible data at each node. As the algorithm progresses, the cost of the provided solution improves, and the expected violation of the power balance constraint decreases. For the asymptotic solution provided by the algorithm, we formally prove an upper bound on the violation of the power balance constraint. We then apply the developed approach to a simulation scenario with a large number of devices. We perform realistic simulations by using open-source historical data, taken from the EU platform Open Power System Data [22], on energy generation/consumption in South Italy. We train a Generative Adversarial Network (GAN) based on these data and use it to generate sample energy generation/consumption profiles. The generated data are used to perform a Monte Carlo numerical experiment on the Italian HPC CINECA infrastructure to show the efficacy of the distributed algorithm. Throughout this article, we assume that primary and secondary controls of the microgrid are already performed by some suitably designed low-level controllers. Thus, we implicitly assume that the stability of the microgrid is maintained through

such controllers. We, instead, aim at computing the power set-points at a slower time scale, i.e., we focus on tertiary control.

Compared to the works [20], [21], the algorithm proposed in this article is also based on a primal decomposition approach as [21]; however, different from these two approaches, our algorithm deals with a stochastic problem, can handle unbounded constraint sets, and does not employ a restriction mechanism. Note that Camisa *et al.* [21] employ suitable optimization variables to handle violations of the coupling constraints. In this article, we will reinterpret this idea by considering similar variables that will instead play the role of second-stage variables in the stochastic scenario.

This article is organized as follows. In Section II, we describe the mixed-integer microgrid model and the stochastic optimal control problem. In Section III, we reformulate the problem as a distributed MILP and apply the two-stage stochastic programming approach. In Section IV, we describe the proposed distributed algorithm and provide theoretical results on the worst case constraint violation, while, in Section V, we discuss Monte Carlo numerical simulations on a practical scenario with a large number of devices and realistic synthesized data.

II. STOCHASTIC MIXED-INTEGER MICROGRID CONTROL WITH RENEWABLES

Let us begin by introducing the mixed-integer microgrid model. For ease of exposition, we consider a fairly general model inspired by the one in [13] without taking into account some specific aspects (see Remark 1). This allows us to better highlight the main features of the proposed approach while keeping the discussion not too technical. Compared to other approaches, the main advantage of using the mixed-integer framework is that it allows to efficiently model logical statements [23] typically arising in microgrids. A microgrid consists of N units, partitioned as follows. Storages are collected in \mathbb{I}_{STOR} , generators in \mathbb{I}_{GEN} , renewable energy sources in \mathbb{I}_{REN} , critical loads in \mathbb{I}_{LO} , controllable loads in \mathbb{I}_{CL} , and one connection with the utility grid in \mathbb{I}_{GRID} . In particular, we consider the point of common coupling to have index $\mathbb{I}_{\text{GRID}} = \{N\}$, while all the other sets are subsets of the full set $\{1, \dots, N\}$. Therefore, the whole set of units can be written as

$$\mathbb{I} = \{1, \dots, N\} = \mathbb{I}_{\text{STOR}} \cup \mathbb{I}_{\text{GEN}} \cup \mathbb{I}_{\text{REN}} \cup \mathbb{I}_{\text{LO}} \cup \mathbb{I}_{\text{CL}} \cup \mathbb{I}_{\text{GRID}}$$

where the sets \mathbb{I}_{STOR} , \mathbb{I}_{GEN} , \mathbb{I}_{REN} , \mathbb{I}_{LO} , \mathbb{I}_{CL} , and \mathbb{I}_{GRID} are pairwise disjoint. Throughout the document, we interchangeably refer to the units also as *agents*. In Sections II-A–II-E, we describe each type of unit separately, while, in Section II-F, we will introduce the optimal control problem. In the following, we denote the optimal control prediction horizon as $K \in \mathbb{N}$.

A. Storages

For storage units $i \in \mathbb{I}_{\text{STOR}}$, let $z_i(k) \in \mathbb{R}$ be the stored energy level at time k , and let $u_i(k) \in \mathbb{R}$ denote the power exchanged with the storage unit at time k (positive for charging and negative for discharging). The dynamics at each time k amounts to $z_i(k+1) = z_i(k) + \eta_i u_i(k) - z_i^{\text{PL}}$, where η_i

TABLE I
LIST OF THE MAIN SYMBOLS AND THEIR DEFINITIONS

Basic definitions	
$N \in \mathbb{N}$	Number of units in the system
$\mathbb{I} = \{1, \dots, N\}$	Set of units
$K > 0$	Prediction horizon of optimal control
$\varepsilon > 0$	Very small number (e.g. machine precision)
Storages (indexed by $i \in \mathbb{I}_{\text{STOR}}$)	
$z_i(k)$	State of charge at time k
$u_i(k)$	Exchanged power (≥ 0 if charging) at time k
$\delta_i(k)$	Charging (1) / discharging (0) state
$\gamma_i(k)$	Auxiliary optimization variable
η_i^c, η_i^d	Charging and discharging efficiencies
$x_i^{\text{MIN}}, x_i^{\text{MAX}}$	Minimum and maximum storage level
x_i^{PL}	Physiological loss of energy
C_i	Maximum output power
ζ_i	Operation and maintenance cost coefficient
Generators (indexed by $i \in \mathbb{I}_{\text{GEN}}$)	
$u_i(k)$	Generated power (≥ 0) at time k
$\delta_i(k)$	On (1) / off (0) state ("on" iff $u_i(k) > 0$)
$\nu_i(k)$	Epigraph variable for quadratic generation cost
$\theta_i^{\text{U}}(k), \theta_i^{\text{D}}(k)$	Epigraph variables for startup/shutdown costs
$T_i^{\text{UP}}, T_i^{\text{DOWN}}$	Minimum up/down time
$u_i^{\text{MIN}}, u_i^{\text{MAX}}$	Min. and max. power that can be generated
r_i^{MAX}	Maximum ramp-up/ramp-down
$\kappa_i^{\text{U}}(k), \kappa_i^{\text{D}}(k)$	Startup and shutdown costs
ζ_i	Operation and maintenance cost coefficient
Renewable energy sources (indexed by $i \in \mathbb{I}_{\text{REN}}$)	
$P_i(k)$	Generated power at time k
Controllable loads (indexed by $i \in \mathbb{I}_{\text{CL}}$)	
$\beta_i(k)$	Curtailment factor ($\in [\beta_i^{\text{MIN}}, \beta_i^{\text{MAX}}$] at time k)
$D_i(k)$	Consumption forecast at time k
$\beta_i^{\text{MIN}}, \beta_i^{\text{MAX}}$	Minimum and maximum allowed curtailment
Connection to the main grid (indexed by $i \in \mathbb{I}_{\text{GRID}}$)	
$u_i(k)$	Imported power from the grid at time k
$\delta_i(k)$	Importing (1) or exporting (0) mode at time k
$\phi_i(k)$	Total expenditure for imported power at time k
$\phi_i^{\text{P}}(k), \phi_i^{\text{S}}(k)$	Price for power purchase and sell at time k
P_i^{MAX}	Maximum exchangeable power
Two-stage stochastic problem	
q_+, q_-	Costs associated to positive and negative recourse

denotes the (dis)charging efficiency and z_i^{PL} is a physiological loss of energy. It is assumed that $\eta_i = \eta_i^c$ if $u_i(k) \geq 0$ (charging mode), whereas $\eta_i = 1/\eta_i^d$ if $u_i(k) < 0$ (discharging mode) with $0 < \eta_i^c$ and $\eta_i^d < 1$. Thus, the dynamics is piecewise linear. To deal with this, we utilize mixed-integer inequalities [23]. Let us introduce additional variables $\delta_i(k) \in \{0, 1\}$ and $\gamma_i(k) \triangleq \delta_i(k)u_i(k) \in \mathbb{R}$ for all k . Each $\delta_i(k)$ is one if and only if $u_i(k) \geq 0$ (i.e., the storage unit at time k is in the charging state). After following the manipulations proposed in [12], we obtain the following model for the i th storage unit:

$$z_i(k+1) = z_i(k) + \left(\eta_i^c - \frac{1}{\eta_i^d} \right) \gamma_i(k) + \frac{1}{\eta_i^d} u_i(k) - z_i^{\text{PL}} \quad (1a)$$

$$E_i^1 \delta_i(k) + E_i^2 \gamma_i(k) \leq E_i^3 u_i(k) + E_i^4 \quad (1b)$$

$$z_i^{\text{MIN}} \leq z_i(k) \leq z_i^{\text{MAX}} \quad (1c)$$

for all time instants k , and

$$z_i(0) = z_{i,0} \quad (1d)$$

where (1a) is the dynamics, (1b) gives mixed-integer inequalities expressing the logical constraints, (1c) gives box constraints on the state of charge (with $0 < z_i^{\text{MIN}} < z_i^{\text{MAX}}$), and (1d) imposes the initial condition ($z_{i,0} \in \mathbb{R}$ is the initial state of charge of storage i). The matrices in (1b) are

$$E_i^1 = \begin{bmatrix} C_i \\ -(C_i + \varepsilon) \\ C_i \\ C_i \\ -C_i \\ -C_i \end{bmatrix}, \quad E_i^2 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \\ 1 \\ -1 \end{bmatrix}$$

$$E_i^3 = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \\ 0 \\ 0 \end{bmatrix}, \quad E_i^4 = \begin{bmatrix} C_i \\ -\varepsilon \\ C_i \\ C_i \\ 0 \\ 0 \end{bmatrix}$$

where $C_i > 0$ is the limit output power and $\varepsilon > 0$ is a very small number (typically machine precision). To each storage i is associated an operation and maintenance cost, which is equal to

$$J_i = \sum_{k=0}^{K-1} \zeta_i |u_i(k)| = \sum_{k=0}^{K-1} \zeta_i (2\gamma_i(k) - u_i(k)) \quad (2)$$

where $\zeta_i > 0$ is the operation and maintenance cost per exchanged unit of power and $2\gamma_i(k) - u_i(k) = |u_i(k)|$ is the absolute value of the power exchanged with the storage.

B. Generators

For generators $i \in \mathbb{I}_{\text{GEN}}$, let $u_i(k) \in \mathbb{R}$ and $u_i(k) \geq 0$ denote the generated power at time k . Since generators can be either ON or OFF, as done for the storages, we let $\delta_i(k) \in \{0, 1\}$ be an auxiliary variable that is equal to 1 if and only if $u_i(k) > 0$. As in the case of storages, we must consider constraints on the operating conditions of generators. Namely, if a generator is turned on/off, there is a minimum amount of time for which the unit must be kept ON/OFF. This logical constraint is modeled by the inequalities

$$\delta_i(k) - \delta_i(k-1) \leq \delta_i(\tau) \quad (3a)$$

$$\tau = k+1, \dots, \min(k + T_i^{\text{UP}} - 1, K)$$

$$\delta_i(k-1) - \delta_i(k) \leq \delta_i(\tau) \quad (3b)$$

$$\tau = k+1, \dots, \min(k + T_i^{\text{DOWN}} - 1, K)$$

for all time instants k , where T_i^{UP} and T_i^{DOWN} are the minimum up and down time of generator i . The power flow limit and the ramp-up/ramp-down limits are modeled, respectively, by

$$u_i^{\text{MIN}} \delta_i(k) \leq u_i(k) \leq u_i^{\text{MAX}} \delta_i(k) \quad (3c)$$

$$-r_i^{\text{MAX}} \delta_i(k) \leq u_i(k) - u_i(k-1) \leq r_i^{\text{MAX}} \delta_i(k) \quad (3d)$$

for all times k , where $u_i^{\text{MAX}} \geq u_i^{\text{MIN}} \geq 0$ denote the maximum and minimum powers that can be generated by generator i and $r_i^{\text{MAX}} \geq 0$ denotes the maximum ramp-up/ramp-down.

The cost associated with generator units is composed of three parts, which are: 1) a (quadratic) generation cost;

2) a startup/shut-down cost; and 3) an operation and maintenance cost. To model the generation cost, we consider a piecewise linearized version $\max_{\ell}(S_i^{\ell}u_i(k) + s_i^{\ell})$ for all k with appropriately defined $S_i^{\ell}, s_i^{\ell} \in \mathbb{R}$. The startup θ_i^U and shutdown cost θ_i^D at each time $k \in \{0, \dots, K-1\}$ are equal to

$$\begin{aligned}\theta_i^U(k) &= \max\{0, \kappa_i^U(k)[\delta_i(k) - \delta_i(k-1)]\} \\ \theta_i^D(k) &= \max\{0, \kappa_i^D(k)[\delta_i(k-1) - \delta_i(k)]\}\end{aligned}$$

where $\kappa_i^U(k), \kappa_i^D(k) > 0$ are the startup and shut-down costs at time k . The operation and maintenance costs are equal to $\zeta_i \delta_i(k)$, where $\zeta_i > 0$ is a cost coefficient (we assume that there is no cost when the generator is turned off). Thus, the expression for the cost of each generator i is

$$J_i = \sum_{k=0}^{K-1} \left[\max_{\ell} (S_i^{\ell} u_i(k) + s_i^{\ell}) + \zeta_i \delta_i(k) + \theta_i^U(k) + \theta_i^D(k) \right].$$

Note that the cost function has internal maximizations and, as such, is nonlinear. However, since the cost is to be minimized, it can be recast as a linear function by introducing so-called epigraph variables (see [24]) as follows. As regards the generation cost, we replace it with epigraph variables $v_i(k) \in \mathbb{R}$ and impose the constraints

$$v_i(k) \geq S_i^{\ell} u_i(k) + s_i^{\ell} \quad \forall \ell \quad (3e)$$

for all times k . Similarly, we can treat $\theta_i^U, \theta_i^D \in \mathbb{R}$ as epigraph variables and write the constraints

$$\theta_i^U(k) \geq \kappa_i^U(k)[\delta_i(k) - \delta_i(k-1)] \quad (3f)$$

$$\theta_i^D(k) \geq \kappa_i^D(k)[\delta_i(k-1) - \delta_i(k)] \quad (3g)$$

$$\theta_i^U(k) \geq 0 \quad (3h)$$

$$\theta_i^D(k) \geq 0 \quad (3i)$$

for all k . We, therefore, obtain the following expression for the cost function of generator i :

$$J_i = \sum_{k=0}^{K-1} [v_i(k) + \theta_i^U(k) + \theta_i^D(k) + \zeta_i \delta_i(k)]. \quad (4)$$

C. Renewable Energy Sources

We consider two types of renewables, namely, wind generators and solar generators. Rather than using a physical or dynamical model for these generators, we use a predictor to generate realistic power production scenarios. Indeed, thanks to the huge amount of historical datasets freely available on the internet, neural network-based predictors have excellent accuracy. More details regarding the scenario generation technique are in Section V-A. We will employ this technique also to generate power demand predictions. These units only contribute to the power balance constraint (9) through their generated power at each time slot k , denoted as $P_i(k) \geq 0$, and do not have associated cost or constraints. Note that $P_i(k)$ are unknown beforehand and, at this point of the derivation, must be modeled as stochastic variables having a certain probability distribution. In principle, the variables $P_i(k)$ should be regarded as having a continuous probability distribution. However, from a computational point of view,

it is convenient to approximate the original distribution with a certain number of scenarios. This approach is known as sample average approximation (SAA) and enjoys asymptotic properties, as outlined in [25]. In the subsequent derivation, we will assume that the distribution of P_i has been already discretized to a finite number of scenarios. Then, since we will exploit a two-stage stochastic optimization approach, we will use the scenario generator to obtain realizations of $P_i(k)$. We discuss this aspect more in detail in Section III.

D. Loads

We consider two types of loads, namely, critical loads and controllable loads. For critical loads $i \in \mathbb{I}_{LO}$, we will denote by $D_i(k)$ the consumption forecast at time k , and we assume that it is given. There are no optimization variables (and, thus, cost functions) associated with this kind of unit; however, their consumption must be considered in the power balance (cf. Section II-F).

For controllable loads $i \in \mathbb{I}_{CL}$, let $D_i(k)$ be the consumption forecast at time k , which is assumed to be given. In case the microgrid has energy shortages, the consumption of controllable loads can be curtailed to meet power balance constraints. This is quantified with a curtailment factor $\beta_i(k) \in [\beta_i^{\min}, \beta_i^{\max}]$, where $0 \leq \beta_i^{\min} \leq \beta_i^{\max} \leq 1$ are the bounds on the allowed curtailment. The actual power consumption at time k is, thus, $(1 - \beta_i(k))D_i(k)$, i.e., if $\beta_i(k) = 0$, there is no curtailment. The curtailment factor is an optimization variable and can be freely chosen; thus, in principle, it can be $\beta_i(k) > 0$ for some k (even if there are no energy shortages) if this results in a cost improvement. The following constraint must be imposed:

$$\beta_i^{\min} \leq \beta_i(k) \leq \beta_i^{\max} \quad (5)$$

for all times k . We assume that the microgrid incurs in a cost that is proportional to the total curtailed power thus, the cost function associated with controllable load i is

$$J_i = \sum_{k=0}^{K-1} \varphi_i D_i(k) \beta_i(k) \quad (6)$$

where $\varphi_i > 0$ is a penalty weight.

E. Connection to the Utility Grid

For the connection with the utility grid $i \in \mathbb{I}_{GRID}$, let $u_i(k) \in \mathbb{R}$ denote the imported (exported) power level from (to) the utility grid. We use the convention that imported power at time k is nonnegative $u_i(k) \geq 0$. As before, since the power purchase price is different from the power sell price, we consider auxiliary optimization variables $\delta_i(k) \in \{0, 1\}$ and $\phi_i(k) \in \mathbb{R}$ for all k . The variable $\delta_i(k)$ models the logical statement $\delta_i(k) = 1$ if and only if $u_i(k) \geq 0$ (i.e., power is imported from the utility grid). The variable $\phi_i(k)$ represents the total expenditure (retribution) for imported (exported) energy. Denoting by $\phi_i^P(k), \phi_i^S(k) \geq 0$ the price for power purchase and sell, it holds $\phi_i(k) = \phi_i^P(k)u_i(k)$ if $\delta_i(k) = 1$ and $\phi_i(k) = \phi_i^S(k)u_i(k)$ if $\delta_i(k) = 0$. By denoting by

$P_i^{\text{MAX}} \geq 0$ the maximum exchangeable power, the corresponding mixed-integer inequalities are (cf. [12])

$$E_i^1 \delta_i(k) + E_i^2 \phi_i(k) \leq E_i^3(k) u_i(k) + E_i^4 \quad (7)$$

for all k , where the matrices are defined as

$$E_i^1 = \begin{bmatrix} P_i^{\text{MAX}} \\ -P_i^{\text{MAX}} - \varepsilon \\ M_i \\ M_i \\ -M_i \\ -M_i \end{bmatrix}, \quad E_i^2 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \\ 1 \\ -1 \end{bmatrix}$$

$$E_i^3(k) = \begin{bmatrix} 1 \\ -1 \\ \phi^p(k) \\ -\phi^p(k) \\ \phi^s(k) \\ -\phi^s(k) \end{bmatrix}, \quad E_i^4 = \begin{bmatrix} P_i^{\text{MAX}} \\ -\varepsilon \\ M_i \\ M_i \\ 0 \\ 0 \end{bmatrix}$$

with $M_i = P_i^{\text{MAX}} \cdot \max_k(\phi^p(k), \phi^s(k))$. It is clear that the cost associated with this unit is

$$J_i = \sum_{k=0}^{K-1} \phi_i(k). \quad (8)$$

F. Power Balance Constraint and Optimal Control Problem

Electrical balance must be met at each time k , i.e.,

$$u_{\text{GRID}}(k) = \sum_{i \in \mathbb{I}_{\text{STOR}}} u_i(k) - \sum_{i \in \mathbb{I}_{\text{GEN}}} u_i(k) + \sum_{i \in \mathbb{I}_{\text{CL}}} (1 - \beta_i(k)) D_i(k) + \sum_{i \in \mathbb{I}_{\text{LO}}} D_i(k) - \sum_{i \in \mathbb{I}_{\text{REN}}} P_i(k). \quad (9)$$

Recall that the length of the prediction horizon is $K \in \mathbb{N}$. The optimal control problem, which is an MILP, can be posed as

$$\min_u \sum_{k=0}^{K-1} \left[\phi_{\text{GRID}}(k) + \sum_{i \in \mathbb{I}_{\text{GEN}}} (\zeta_i \delta_i(k) + v_i(k) + \theta_i^u(k) + \theta_i^d(k)) + \sum_{i \in \mathbb{I}_{\text{CL}}} \varphi_i D_i(k) \beta_i(k) + \sum_{i \in \mathbb{I}_{\text{STOR}}} (\zeta_i (2\gamma_i(k) - u_i(k))) \right]$$

s.t. storage constraints (1)
generator constraints (3)
constraints (5), (7), (9). (10)

In the formulation of problem (10), the presence of the renewables is taken into account by the power balance constraint (9). Indeed, constraint (9) contains the terms $P_i(k)$, $i \in \mathbb{I}_{\text{REN}}$, which represents the power generated by renewable i at the time step k of the prediction horizon. Note that problem (10) is a stochastic optimization problem. Indeed, the equality constraint (9) is stochastic since it depends on $P_i(k)$. Next, we show how to handle this level of complexity.

Remark 1: Note that the microgrid model can also be extended to additionally consider thermal loads and combined heat and power (CHP) units, which would additionally require a thermal balance constraint. The architecture proposed in the following can be easily adapted to deal with this scenario

by making only minor changes. However, in order not to complicate too much the notation, we prefer not to introduce this further level of complexity, which nevertheless can be handled by the proposed framework. \square

Remark 2: An important fact of the model described so far is that, as also done in [13], we explicitly consider the dynamics only for storages in order to keep the computational complexity of the problem low. This is due to the fact that, given the considered time scale, it is not worth considering oscillatory behaviors of generators or of other units, which happens at a much smaller time scale. For the same reason, as regards renewables, we prefer to have an external tool generate a plausible 24-h trajectory based on historical data, rather than modeling all the dynamics of the unit. In this way, essential features of both generators and renewables are appropriately taken into account while keeping a low number of optimization variables. Instead, for storages, we need to explicitly keep track of the amount of stored energy at each time slot. \square

III. DISTRIBUTED CONSTRAINT-COUPLED STOCHASTIC OPTIMIZATION

To handle the stochastic quantities $P_i(k)$, we follow the ideas of Parisio *et al.* [13] and utilize a two-stage stochastic optimization approach. As we are interested in a distributed algorithm, instead of applying the two-stage stochastic approach directly to problem (10), we rather apply it to a distributed reformulation of problem (10). In this section, we first introduce the distributed reformulation of the problem and then formalize the two-stage stochastic optimization approach. The distributed approach to be introduced shortly will allow us to compute a feasible (in general suboptimal) solution to the problem, with guaranteed performance bounds, as discussed in Section IV-B.

A. Constraint-Coupled Reformulation

The optimal control problem (10) can be reformulated in such a way that the distributed structure of the problem becomes more evident. Formally, problem (10) is equivalent to the stochastic *constraint-coupled* MILP

$$\min_{x_1, \dots, x_N} \sum_{i=1}^N c_i^\top x_i$$

$$\text{s.t. } \sum_{i=1}^N A_i x_i = b$$

$$x_i \in X_i^{\text{MILP}}, \quad i = 1, \dots, N \quad (11)$$

where, for all $i \in \{1, \dots, N\}$, the decision vector x_i has $n_i = p_i + q_i$ components (thus, $c_i \in \mathbb{R}^{n_i}$) with $p_i, q_i \in \mathbb{N}$, and the local constraint set is of the form

$$X_i^{\text{MILP}} = P_i \cap (\mathbb{Z}^{p_i} \times \mathbb{R}^{q_i})$$

for some nonempty compact polyhedron $P_i \subset \mathbb{R}^{p_i+q_i}$. Moreover, the matrices $A_i \in \mathbb{R}^{K \times n_i}$ and the vector $b \in \mathbb{R}^K$ model coupling constraints among the variables. We recall from Section II that the set of agents consists of

$\mathbb{I} = \{1, \dots, N\} = \mathbb{I}_{\text{STOR}} \cup \mathbb{I}_{\text{GEN}} \cup \mathbb{I}_{\text{REN}} \cup \mathbb{I}_{\text{LO}} \cup \mathbb{I}_{\text{CL}} \cup \mathbb{I}_{\text{GRID}}$. The term “constraint-coupled” that we associate to problem (11) is due to the fact that the constraints $\sum_{i=1}^N A_i x_i = b$ create a link among all the variables x_1, \dots, x_N , which otherwise could be optimized independently of each other. To achieve the mentioned reformulation, we now specify the quantities x_i , c_i , X_i^{MILP} , and A_i for each type of device and the right-hand side vector b .

1) *Storages*: We assume that each storage unit $i \in \mathbb{I}_{\text{STOR}}$ is responsible for the optimization vector x_i consisting of the stack of $z_i(k), u_i(k), \gamma_i(k) \in \mathbb{R}$, and $\delta_i(k) \in \{0, 1\}$ for all $k \in \{0, \dots, K-1\}$ plus the variable $z_i(K) \in \mathbb{R}$. The constraints in X_i^{MILP} are given by (1), while the cost function is $c_i^\top x_i = \sum_{k=0}^{K-1} \zeta_i(2\gamma_i(k) - u_i(k))$.

2) *Generators*: Each generator $i \in \mathbb{I}_{\text{GEN}}$ is responsible for the optimization vector x_i consisting of the stack of $u_i(k), v_i(k), \theta_i^u(k), \theta_i^d(k) \in \mathbb{R}$, and $\delta_i(k) \in \{0, 1\}$ for all $k \in \{0, \dots, K-1\}$. The constraints in X_i^{MILP} are given by (3a)–(3i), while the cost function is $c_i^\top x_i = \sum_{k=0}^{K-1} (\zeta_i \delta_i(k) + v_i(k) + \theta_i^u(k) + \theta_i^d(k))$.

3) *Critical Loads*: For the critical loads $i \in \mathbb{I}_{\text{LO}}$, there are no variables to optimize, but they must be taken into account in the coupling constraints.

4) *Controllable Loads*: For each controllable load $i \in \mathbb{I}_{\text{CL}}$, the optimization vector x_i consists of the stack of $\beta_i(k) \in \mathbb{R}$, for all $k \in \{0, \dots, K-1\}$, with constraints given by (5). Note that, for this class of devices, the local constraint set is not mixed-integer. The cost function is $c_i^\top x_i = \sum_{k=0}^{K-1} \varphi_i D_i(k) \beta_i(k)$.

5) *Connection to the Utility Grid*: For this device $i \in \mathbb{I}_{\text{GRID}}$, the optimization vector x_i consists of the stack of $u_i(k), \phi_i(k) \in \mathbb{R}$ and $\delta_i(k) \in \{0, 1\}$ for all $k \in \{0, \dots, K-1\}$. The local constraints are (7), while the cost function is $c_i^\top x_i = \sum_{k=0}^{K-1} \phi_i(k)$.

6) *Coupling Constraints*: Finally, the coupling constraints are given by (9), which can be encoded in the form $\sum_{i=1}^N A_i x_i = b$ by appropriately defining the matrices A_i and the vector b . In particular, the matrices $A_i \in \mathbb{R}^{K \times n_i}$ are such that

$$[A_i x_i]_k = u_i(k) \quad \forall i \in \mathbb{I}_{\text{STOR}} \quad (12a)$$

$$[A_i x_i]_k = -u_i(k) \quad \forall i \in \mathbb{I}_{\text{GEN}} \quad (12b)$$

$$[A_i x_i]_k = -\beta_i(k) D_i(k) \quad \forall i \in \mathbb{I}_{\text{CL}} \quad (12c)$$

$$[A_i x_i]_k = -u_i(k) \quad \forall i \in \mathbb{I}_{\text{GRID}} \quad (12d)$$

for all times k , while the right-hand side vector $b \in \mathbb{R}^K$ is equal to

$$b = - \sum_{i \in \mathbb{I}_{\text{CL}}} D_i - \sum_{i \in \mathbb{I}_{\text{LO}}} D_i + \sum_{i \in \mathbb{I}_{\text{REN}}} P_i \quad (12e)$$

where $D_i \in \mathbb{R}^K$ and $P_i \in \mathbb{R}^K$ denote the stack of $D_i(k)$ and $P_i(k)$ for all times k . Note that the power generated by the renewables introduces a stochasticity in the right-hand side vector b appearing in problem (11).

Remark 3: It should be noted that, thanks to its general formulation, problem (11) allows for the presence of new units in the grid, which can be always added and treated as new agents. \square

In the considered distributed context, we assume that each agent i does not know the entire problem information. In particular, we assume that it only knows the local cost vector c_i , the local constraint X_i^{MILP} , and its matrix A_i of the coupling constraint. The exchange of information among N agents occurs according to a graph-based communication model. We use $\mathcal{G} = (V, \mathcal{E})$ to indicate the undirected, connected graph describing the network, where $V = \{1, \dots, N\}$ is the set of vertices and \mathcal{E} is the set of edges. If $(i, j) \in \mathcal{E}$, then agent i can communicate with agent j and vice versa. We use \mathcal{N}_i to indicate the set of neighbors of agent i in \mathcal{G} , i.e., $\mathcal{N}_i = \{j \in V \mid (i, j) \in \mathcal{E}\}$.

B. Two-Stage Stochastic Optimization Approach

In its current form, problem (11) cannot be practically solved due to the right-hand side vector b being unknown. To deal with this, the approach consists of considering a set of possible *scenarios* that may arise and then to formulate and solve a so-called *two-stage* stochastic optimization problem, which we now introduce. The scheme described in the next is inspired by the approach introduced in [13], which we suitably adapt in order to deal with our distributed scenario.

Intuitively, in this uncertain scenario, one has to “*a priori*” (i.e., without knowing the actual value of the random vector b) choose a set of control actions $u_i(k)$, such as generated/stored power or power curtailments, in order to minimize a certain cost criterion in an expected sense. However, these control actions will inevitably result in a violation of the power balance constraint (9) “*a posteriori*” (i.e., when the actual power production of renewables, and hence the value of the random vector b , becomes known). To compensate for this infeasibility, *recourse* actions must be taken. These actions are associated with a cost and will have an impact on the final performance achieved by the whole control scheme. In the jargon of two-stage stochastic optimization, the first-stage optimization variables are those associated with the control actions [i.e., x_1, \dots, x_N in problem (11)], while the second-stage optimization variables (to be introduced shortly) are those associated with recourses.

Formally, we denote by ω the random vector collecting all the renewable energy generation profiles. We assume a finite discrete probability distribution for ω , and we denote by π_r the probability of each ω_r , i.e., $\pi_r = \mathbb{P}(\omega = \omega_r)$ for all $r \in \{1, \dots, R\}$. To keep the notation consistent, we denote the renewable energy profile corresponding to ω_r as $P_{ir}(k)$. We denote by b_r the realization of b associated with the scenario ω_r . Using these positions, the two-stage stochastic MILP can be formulated as (see [26])

$$\begin{aligned} \min_{x_1, \dots, x_N} \quad & \sum_{i=1}^N c_i^\top x_i + \mathbb{E}_\omega [Q(x_1, \dots, x_N, \omega)] \\ \text{s.t.} \quad & x_i \in X_i^{\text{MILP}}, \quad i = 1, \dots, N \end{aligned} \quad (13)$$

where x_1, \dots, x_N are the first-stage variables modeling the (*a priori*) control actions and $Q(x_1, \dots, x_N, \omega)$ is the optimal

value of the second-stage problem

$$\begin{aligned} Q(x_1, \dots, x_N, \omega) = \min_{\eta_+^\omega, \eta_-^\omega} & \sum_{k=0}^{K-1} (q_+ \eta_{k+}^\omega + q_- \eta_{k-}^\omega) \\ \text{s.t.} & -\eta_{r-}^\omega \leq \sum_{i=1}^N A_i x_i - b_\omega \leq \eta_{r+}^\omega \\ & \eta_+^\omega, \eta_-^\omega \geq 0. \end{aligned} \quad (14)$$

Here, $\eta_+^\omega, \eta_-^\omega \in \mathbb{R}^K$ are the two-stage variables modeling the (*a posteriori*) recourse actions, which are penalized in the cost with $q_+ \geq 0$ and $q_- \geq 0$, which are the costs related to energy surplus and shortage, respectively. Thus, the function $Q(x_1, \dots, x_N, \omega_r)$ appearing in problem (13) models the cost of the recourse when the first-stage variables are chosen as x_1, \dots, x_N and the realization of the random variable is a certain ω_r . Since the distribution of ω is discrete, it holds

$$\mathbb{E}_\omega[Q(x_1, \dots, x_N, \omega)] = \sum_{r=1}^R \pi_r Q(x_1, \dots, x_N, \omega_r) \quad (15)$$

and, thus, we can reformulate the stochastic problem (13) as the equivalent deterministic formulation

$$\begin{aligned} \min_{\substack{x_1, \dots, x_N \\ \eta_+, \eta_-}} & \sum_{i=1}^N c_i^\top x_i + \sum_{k=0}^{K-1} \sum_{r=1}^R \pi_r (q_+ \eta_{kr+} + q_- \eta_{kr-}) \\ \text{s.t.} & -\eta_{r-} \leq \sum_{i=1}^N A_i x_i - b_r \leq \eta_{r+}, \quad r = 1, \dots, R \\ & \eta_+, \eta_- \geq 0 \\ & x_i \in X_i^{\text{MILP}}, \quad i = 1, \dots, N. \end{aligned} \quad (16)$$

In problem (16), we denoted by η_{kr+} the variable associated with positive recourse for scenario r at time k . We also use the symbol η_{r+} to denote the stack of η_{kr+} for all k . The stack of η_{kr+} for all k and r is denoted by η_+ . A similar notation holds for η_- .

At a first glance, it may seem that the two-stage problem (16) loses the constraint-coupled structure of the distributed optimization problem (11). However, with a bit of manipulation, it is still possible to arrive at a similar result. We begin by streamlining the notation. Define $\eta \in \mathbb{R}^{2KR}$, $\eta \geq 0$ as the stack of η_+ and η_- , and the vector $d \in \mathbb{R}^{2KR}$ such that $d^\top \eta = \sum_{k=0}^{K-1} \sum_{r=1}^R \pi_r (q_+ \eta_{kr+} + q_- \eta_{kr-})$. Moreover, define $H_i \in \mathbb{R}^{2KR \times n_i}$ and $h \in \mathbb{R}^{2KR}$ with

$$\begin{aligned} H_i &= \mathbf{1} \otimes [A_i^\top \quad -A_i^\top]^\top = [A_i^\top \quad -A_i^\top \quad \dots \quad A_i^\top \quad -A_i^\top]^\top \\ h &= [b_1^\top \quad -b_1^\top \quad \dots \quad b_R^\top \quad -b_R^\top]^\top \end{aligned}$$

where $\mathbf{1} \in \mathbb{R}^R$ is the vector of ones and \otimes denotes the Kronecker product. Thus, problem (16) is equivalent to

$$\begin{aligned} \min_{\substack{x_1, \dots, x_N \\ \eta}} & \sum_{i=1}^N c_i^\top x_i + d^\top \eta \\ \text{s.t.} & \sum_{i=1}^N H_i x_i - h \leq \eta \\ & \eta \geq 0, \quad x_i \in X_i^{\text{MILP}}, \quad i = 1, \dots, N. \end{aligned} \quad (17)$$

By defining $\eta_1, \dots, \eta_N \in \mathbb{R}^{2RK}$ such that $\sum_{i=1}^N \eta_i = \eta$ and each $\eta_i \geq 0$, we see that problem (17) is, finally, equivalent to

$$\begin{aligned} \min_{\substack{x_1, \dots, x_N \\ \eta_1, \dots, \eta_N}} & \sum_{i=1}^N (c_i^\top x_i + d^\top \eta_i) \\ \text{s.t.} & \sum_{i=1}^N (H_i x_i - \eta_i) \leq h \\ & \eta_i \geq 0, \quad x_i \in X_i^{\text{MILP}}, \quad i = 1, \dots, N \end{aligned} \quad (18)$$

in the sense that any solution of (17) can be reconstructed from a solution of (18) by using $\eta = \sum_{i=1}^N \eta_i$. Note that problem (18) has an unbounded feasible set (because of the variables η_i), but it always admits an optimal solution due to the terms $d^\top \eta_i$ minimized in the cost (recall that $d \geq 0$).

IV. DISTRIBUTED ALGORITHM AND ANALYSIS

We now propose a distributed algorithm to compute a feasible solution to problem (18) and provide the convergence results.

A. Distributed Algorithm Description

Let us begin by describing the proposed distributed algorithm to solve problem (18). The basic idea behind the distributed algorithm is to compute a mixed-integer solution starting from an optimal solution of the convex relaxation of problem (17) obtained by replacing X_i^{MILP} with their convex hull $\text{conv}(X_i^{\text{MILP}})$

$$\begin{aligned} \min_{\substack{z_1, \dots, z_N \\ \eta_1, \dots, \eta_N}} & \sum_{i=1}^N (c_i^\top z_i + d^\top \eta_i) \\ \text{s.t.} & \sum_{i=1}^N (H_i z_i - \eta_i) \leq h \\ & \eta_i \geq 0, \quad z_i \in \text{conv}(X_i^{\text{MILP}}), \quad i = 1, \dots, N \end{aligned} \quad (19)$$

where we denote by z_i the continuous counterpart of the mixed-integer variable x_i . To do so, each agent i maintains an auxiliary variable $y_i^t \in \mathbb{R}^{2RK}$, which represents a local *allocation* of the coupling constraints (cf. Appendix A). At each iteration t , the vector y_i^t is updated according to (20) and (21). After $T_f > 0$ iterations, the agent computes a tentative mixed-integer solution based on the last computed allocation estimate [cf. (22)]. Algorithm 1 summarizes the steps from the perspective of agent i .

Let us briefly comment on the algorithm structure. As it will be clear from the analysis, the first two steps (20) and (21) are used to compute an optimal solution of problem (19), while the last step (22) reconstructs a mixed-integer solution. Note that problem (20) is an LP, and problem (22) is an MILP. From a computational point of view, in order to solve the optimization problems appearing in Algorithm 1, a Lagrange multiplier of problem (20) can be found by each agent by running either a dual subgradient method or a dual cutting-plane method (cf. [21]), while an optimal solution to problem (22) can be found with any MILP solver. As regards the computational complexity of the algorithm, we highlight that, due

Algorithm 1 Distributed Stochastic Mixed-Integer Microgrid Control

Initialization: set $T_f > 0$ and y_i^0 such that $\sum_{i=1}^N y_i^0 = h$

Repeat for $t = 0, 1, \dots, T_f - 1$

Compute μ_i^t as a Lagrange multiplier of

$$\begin{aligned} \min_{z_i, \eta_i} \quad & c_i^\top z_i + d^\top \eta_i \\ \text{subj. to} \quad & \mu_i : H_i z_i \leq y_i^t + \eta_i \\ & \eta_i \geq 0, \quad z_i \in \text{conv}(X_i^{\text{MILP}}) \end{aligned} \quad (20)$$

Receive μ_j^t from $j \in \mathcal{N}_i$ and update

$$y_i^{t+1} = y_i^t + \alpha^t \sum_{j \in \mathcal{N}_i} (\mu_i^t - \mu_j^t) \quad (21)$$

Return $(x_i^{T_f}, \eta_i^{T_f})$ as optimal solution of

$$\begin{aligned} \min_{x_i, \eta_i} \quad & c_i^\top x_i + d^\top \eta_i \\ \text{subj. to} \quad & H_i x_i \leq y_i^{T_f} + \eta_i \\ & \eta_i \geq 0, \quad x_i \in X_i^{\text{MILP}} \end{aligned} \quad (22)$$

to the update (21), the amount of computation performed by each agent grows linearly with the number of neighbors $|\mathcal{N}_i|$ and is independent of the overall number of agents N . In Section IV-B, we will prove a worst case violation of the power balance constraints.

Remark 4: An important fact is that the computed mixed-integer solution always satisfies the coupling constraint appearing in problem (17) with a possibly high η_i , i.e.,

$$\sum_{i=1}^N (H_i x_i^{T_f} - \eta_i^{T_f}) \leq \sum_{i=1}^N y_i^{T_f} = h$$

where the inequality follows by construction and the equality follows by the forthcoming Lemma 2. Thus, the algorithm can be stopped at any iteration $T_f \geq 0$, and the resulting solution will be feasible for the two-stage MILP (17). The greater the number of iterations, the higher is the optimality of the computed solution, and the lower is the expected violation of the original power balance constraint. \square

B. Theoretical Results

In this subsection, we provide theoretical results on Algorithm 1. In particular, we will prove a bound for the worst case violation of the asymptotically computed mixed-integer solution. Indeed, as we will prove shortly, the algorithm enjoys asymptotic convergence; however, it also provides a feasible solution at each iteration, and therefore, it can be safely stopped prematurely. We will then consider the coupling constraints $\sum_{i=1}^N H_i x_i - h \leq \eta$ appearing in problem (17), and we will derive an upper bound for the value of η . In other words, we will derive an upper bound for the maximum component of the vector $\sum_{i=1}^N H_i x_i - h$, where the maximum is taken with respect to both the prediction horizon and the scenarios.

To begin with, we recall some preliminary lemmas, where we remind that K denotes the prediction horizon and R is the total number of scenarios in the stochastic problems. We highlight that the results to be proved next are true regardless of the value of K and R . As a matter of fact, we assume that these parameters have been fixed *a priori* appropriately.

Lemma 1 [21]: Let problem (19) be feasible, and let $(\bar{z}_1, \dots, \bar{z}_N, \bar{\eta}_1, \dots, \bar{\eta}_N)$ be any vertex of its feasible set. Then, there exists an index set $I_{\bar{z}} \subseteq \{1, \dots, N\}$, with cardinality $|I_{\bar{z}}| \geq N - 2RK$, such that $\bar{z}_i \in X_i^{\text{MILP}}$ for all $i \in I_{\bar{z}}$. \square

The consequence of Lemma 1 is that at least $N - 2RK$ blocks of the mixed-integer solution computed asymptotically by Algorithm 1 are equal to the corresponding blocks of optimal solution of (19). Next, we recall convergence of the steps (20) and (21). To this end, we denote as $(z_1^{\text{LP}}, \dots, z_N^{\text{LP}}, \eta_1^{\text{LP}}, \dots, \eta_N^{\text{LP}})$ an optimal solution of problem (19), together with the allocation vector $(y_1^{\text{LP}}, \dots, y_N^{\text{LP}})$ associated with the primal decomposition master problem (cf. Appendix A), which is a vector satisfying

$$H_i z_i^{\text{LP}} - \eta_i^{\text{LP}} \leq y_i^{\text{LP}} \quad \text{for all } i \in \{1, \dots, N\} \quad (23a)$$

and

$$\sum_{i=1}^N y_i^{\text{LP}} = h. \quad (23b)$$

The following assumption is made on the step-size sequence.

Assumption 1: The step-size sequence $\{\alpha^t\}_{t \geq 0}$, with each $\alpha^t \geq 0$, satisfies $\sum_{t=0}^{\infty} \alpha^t = \infty$, $\sum_{t=0}^{\infty} (\alpha^t)^2 < \infty$. \square

The following proposition summarizes the convergence properties of the steps (20) and (21).

Lemma 2 [21]: Let problem (19) be feasible, and let Assumption 1 hold. Consider the allocation vector sequence $\{y_1^t, \dots, y_N^t\}_{t \geq 0}$ generated by steps (20) and (21) of Algorithm 1 with the allocation vectors y_i^0 initialized such that $\sum_{i=1}^N y_i^0 = h$. Then, the following holds.

- 1) $\sum_{i=1}^N y_i^t = h$ for all $t \geq 0$.
- 2) $\lim_{t \rightarrow \infty} \|y_i^t - y_i^{\text{LP}}\| = 0$ for all $i \in \{1, \dots, N\}$. \square

Because of Lemma 2, from now on, we concentrate on the asymptotic mixed-integer solution computed by Algorithm 1. In particular, we denote by $(x_i^\infty, \eta_i^\infty)$ the optimal solution of problem (22) with allocation equal to y_i^{LP} , i.e.,

$$\begin{aligned} \min_{x_i, \eta_i} \quad & c_i^\top x_i + d^\top \eta_i \\ \text{s.t.} \quad & H_i x_i \leq y_i^{\text{LP}} + \eta_i \\ & \eta_i \geq 0, \quad x_i \in X_i^{\text{MILP}}. \end{aligned} \quad (24)$$

We also define the lower bound of resources $\ell_i \in \mathbb{R}^{2RK}$

$$\begin{aligned} \ell_i \triangleq \min_{x_i, \eta_i} \quad & H_i x_i - \eta_i \\ \text{s.t.} \quad & x_i \in \text{conv}(X_i) \\ & 0 \leq \eta_i \leq M \mathbf{1} \end{aligned}$$

where min is componentwise and $M > 0$ is a sufficiently large number. Thus, it holds $\ell_i \leq y_i$ for all admissible allocations y_i , and in particular, $\ell_i \leq y_i^{\text{LP}}$. Operatively, since the constraints on x_i and η_i are disjoint, the vector ℓ_i can be computed by

replacing $x_i \in \text{conv}(X_i)$ with $x_i \in X_i$. In the next theorem, we formalize the bound on the worst case violation.

Theorem 1: Let problem (19) be feasible and consider the asymptotic mixed-integer solution $(x_i^\infty, \eta_i^\infty)$ computed by each agent $i \in \{1, \dots, N\}$. Then, the worst case violation of the power balance constraint is

$$\sum_{i=1}^N H_i x_i^\infty - h \leq \sum_{i \in I_Z} \eta_i^{\text{LP}} + \sum_{i \notin I_Z} \frac{c_i^\top (x_i^{\text{L}} - x_i^\infty) + d^\top \eta_i^{\text{L}}}{d^{\text{MIN}}} \mathbf{1}$$

where $d^{\text{MIN}} = \min_{j \in \{1, \dots, 2RK\}} d_j$, I_Z denotes the set of agents (satisfying $|I_Z| \geq N - 2RK$) for which $z_i^{\text{LP}} \in X_i^{\text{MILP}}$, and $(x_i^{\text{L}}, \eta_i^{\text{L}})$ is an optimal solution of problem (29). \square

The proof is provided in Appendix B. Note that, since this bound is the sum of contributions of the agents, it can be computed *a posteriori* in a distributed way using a consensus scheme. To do so, they first need to detect whether they belong to I_Z or not by computing the primal solution z_i^{LP} of (20) and by checking whether it satisfies $z_i^{\text{LP}} \in X_i^{\text{MILP}}$. Then, they run the consensus scheme using as initial condition either $N\eta_i^{\text{LP}}$ (if $z_i^{\text{LP}} \in X_i^{\text{MILP}}$) or $N((c_i^\top (x_i^{\text{L}} - x_i^\infty) + d^\top \eta_i^{\text{L}})/d^{\text{MIN}})\mathbf{1}$ (if $z_i^{\text{LP}} \notin X_i^{\text{MILP}}$).

V. NUMERICAL EXPERIMENTS

In this section, we validate the proposed framework through large-scale numerical computations. All the simulations are performed with the DISROPT package [27] and are performed on the Italian HPC CINECA infrastructure. In order to make the simulations realistic, we run Algorithm 1 on a generated problem with data synthesized using a deep GAN [28]. In the next subsections, we first provide details regarding the scenario generation for renewable energy sources; then, we show aggregate results on Monte Carlo simulations. Finally, we show in more detail one specific simulation. We point out that the algorithm is specifically tailored for the considered scenario and, as such, cannot be compared numerically with other distributed approaches in the literature for MILPs, such as [20] and [21]. Indeed, both of them would require the constraint sets of the local variables x_i, η_i in problem (18) to be bounded, which does not hold in our case due to the variables $\eta_i \geq 0$.

A. Scenario Generation With Generative Adversarial Networks

Recall that $b \in \mathbb{R}^K$ is a random variable that depends on the total energy produced by the renewables (12e). The variable b has its own probability distribution, and $b_1, \dots, b_R \in \mathbb{R}^K$ are randomly drawn samples [cf. (16)]. In order to generate such samples, we utilize a GAN trained with an open historical dataset from the EU. To train the neural network, we used the data series provided by Open Power System Data [22]. In particular, we used the generation data of renewable energy sources in South Italy. To guarantee a certain uniformity of the data, we narrowed the dataset by concentrating only on summer months and discarded days with missing information. Each sample is a vector in \mathbb{R}^{24} and contains information on the power produced during a day with an hourly resolution.

As for the utilized neural networks, the generative networks have a 10-D input with the following layers:

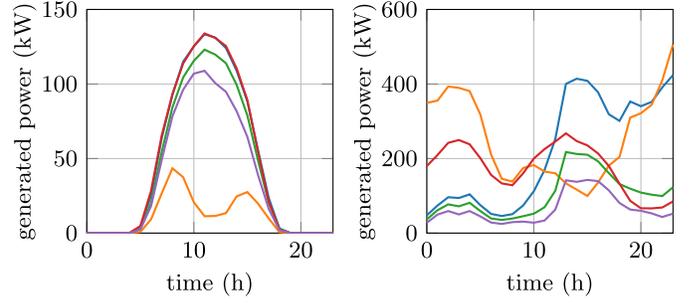


Fig. 1. Five examples of power generation profiles generated by the GANs. Left: solar energy. Right: wind energy.

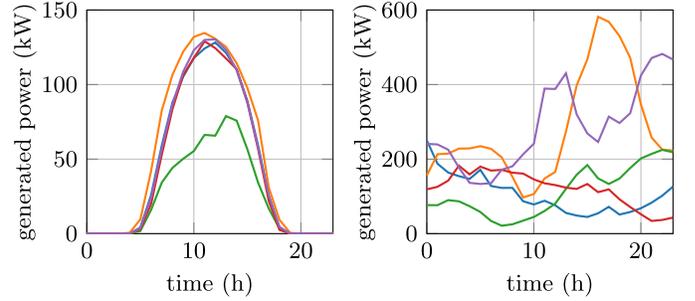


Fig. 2. Five examples of power generation profiles extracted by historical data. Left: solar energy. Right: wind energy.

- 1) a dense layer with 1536 units, batch normalization, and leaky ReLU activation function;
- 2) a layer that reshapes the input to the shape (6, 256);
- 3) a transposed convolution layer with 128 output filters, kernel size equal to 5, stride 1, batch normalization, and leaky ReLU activation function;
- 4) a transposed convolution layer with 64 output filters, kernel size equal to 5, stride 2, batch normalization, and leaky ReLU activation function;
- 5) a transposed convolution layer with 1 output filter, kernel size equal to 5, stride 2, and *tanh* activation function.

The output of the generative network is a 24-D vector containing the power produced by the renewable unit at each time slot of the day. The discriminator networks have a 24-D input with the following layers:

- 1) a convolution layer with 64 output filters, kernel size equal to 5, stride 2, and leaky ReLU activation function;
- 2) a dropout layer with the rate of 0.3;
- 3) a convolution layer with 128 output filters, kernel size equal to 5, stride 2, and leaky ReLU activation function;
- 4) a dropout layer with rate 0.3;
- 5) a layer that flattens the input;
- 6) a dense layer with one output unit.

The output of the discriminator networks is a scalar that denotes the probability that the evaluated input is a real one or a generated one.

We used neural networks to generate samples of solar energy and wind energy. We used TENSORFLOW 2.4 to model the networks and we performed the training with 10^4 epochs using the ADAM algorithm. In Fig. 1, we show example

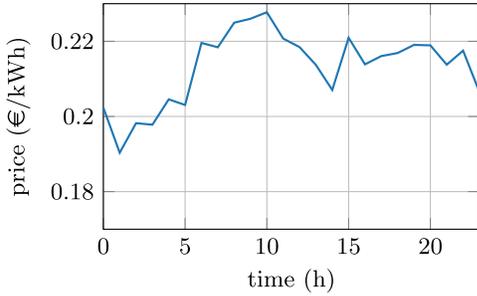


Fig. 3. Daily spot prices from Open Power System Data [22].

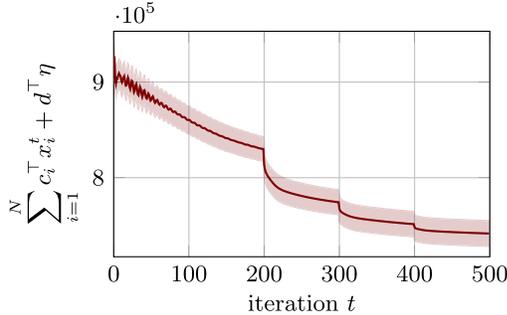


Fig. 4. Evolution of the cost yielded by Algorithm 1. The solid line is the mean value of the Monte Carlo trials, while the dashed area represents one standard deviation.

profiles of solar and wind energy generated by the networks. For comparison, in Fig. 2, we provide examples of profiles extracted from historical data. It can be noted that generated trajectories of solar energy production have a maximum at midday, while one of the trajectories has lower values than the others and may be associated, for instance, with a cloudy day. In any case, the power generated outside the time window 5 A.M.–8 P.M. is close to zero, consistently with real profiles.

B. Monte Carlo Simulations

To test the proposed framework, we performed 100 Monte Carlo simulations in which we run Algorithm 1 on different realizations of the energy generation scenarios (i.e., different realizations of b).

We considered a microgrid control problem with the following units: 20 generators, 20 storages, 60 controllable loads, 20 critical loads, 40 solar generators, 15 wind generators, and the connection to the main grid. For each instance of the problem, we extracted $R = 5$ scenarios and fixed a 24-h prediction horizon and a 1-h sampling time. The initial conditions of storages and generators are generated randomly. As regards the load profiles and the daily spot prices, we utilized the data provided by [22], which are shown in Fig. 3. We then executed Algorithm 1 for 500 iterations with a piecewise constant step size that we initialize to 3.0 and multiply by 0.5 every 100 iterations.

The results of the simulations are shown in Figs. 4 and 5. In Fig. 4, we plot the cost of the mixed-integer solution computed by the algorithm throughout its evolution [in particular, the cost function of the two-stage problem (16)]. The picture highlights how the algorithm improves the cost at each

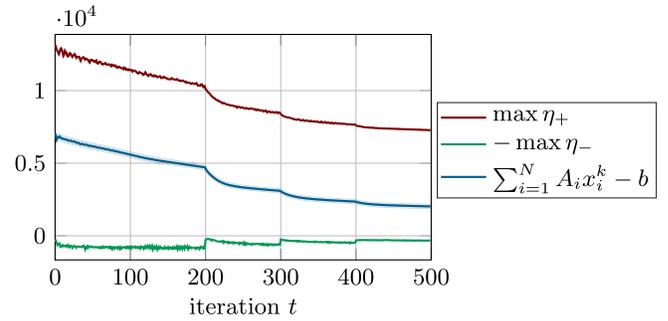


Fig. 5. Evolution of the coupling constraint value throughout the evolution of Algorithm 1. The blue line represents the average value of the power balance constraint (the dashed area corresponds to one standard deviation). The upper and lower lines are the maximum positive and negative two-stage violations of the constraints.

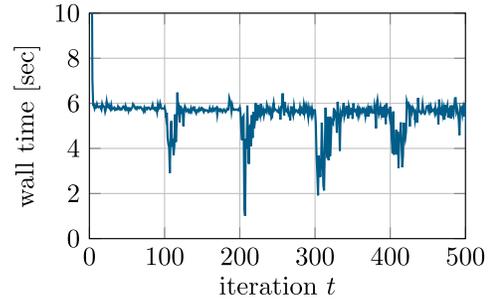


Fig. 6. Computation times when running Algorithm 1 on a single instance of the Monte Carlo simulation.

iteration, i.e., the more the iterations are performed, the more the solution performance improves.

In Fig. 5, we show the value of the coupling constraints for the two-stage problem (16). The red and green lines correspond to the maximum value of η_+ and η_- (with changed sign), respectively, where the maximum is taken with respect to the scenarios, the components of the constraint, and the Monte Carlo trials. The blue line represents the average value of the power balance constraint, while the dashed area corresponds to one standard deviation of the Monte Carlo trials. At each time step, the power balance constraints are always in between the upper and lower lines, while the uncertainty range reduces as the algorithm progresses.

C. Results on a Single Instance

To conclude this section, we show how Algorithm 1 behaves on a single instance of the Monte Carlo trials. To begin with, we perform a computational study of the computation times. Thanks to the fact that we run Algorithm 1 on an HPC, each agent is assigned to a single processor, and thus, computations are really performed in parallel. For each unit involved in the optimization, we recorded the wall time needed for performing the iterations. Fig. 6 represents the computation time per iteration. It can be seen that, even though the problem is large-scale and mixed-integer (cf. the discussion in Section I), each iteration of the distributed algorithm on this large-scale problem takes approximately 6 s.

Now, we provide details regarding the solution computed by Algorithm 1 on a single instance of the Monte Carlo

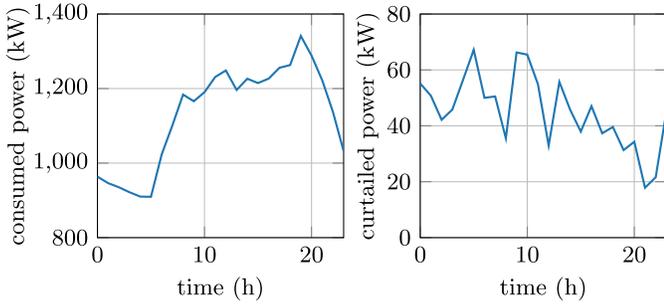


Fig. 7. Total consumed power (critical and controllable loads) and curtailed power (for controllable loads only).

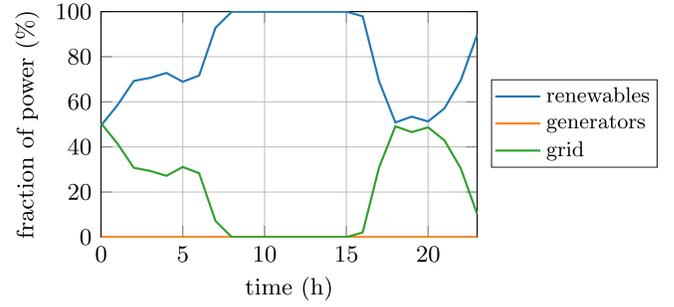


Fig. 10. Fraction of consumed power coming from generators, renewables, and utility grid at each time slot.

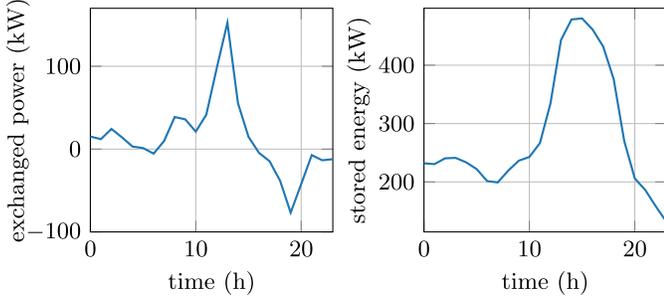


Fig. 8. Total average power exchanged by storage units (left) and level of total stored power (right).

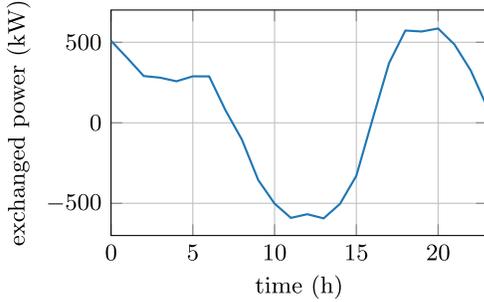


Fig. 9. Total power exchanged with the utility grid.

trials. In Fig. 7, we show the total consumed power and the total curtailed power. In Fig. 8, we show the total power exchanged with storage units (a positive value means that, overall, the storage units are charging) and the global level of stored power. The solution provided by the algorithm is such that storages accumulate as much energy as they can during the peaks of power produced by the renewables. This energy is then released during the subsequent hours of the day. In Fig. 9, we show the total power exchanged with the utility grid (a positive value means that power is purchased from the grid). Note that, during the peak of power produced by the renewables, the microgrid exports energy to the main grid in order to maximize the income. In Fig. 10, we show where does the total available power come from. In particular, we highlight the fraction of power coming from generators, renewables, and the utility grid. In this simulation, the generators did not produce any energy.

VI. CONCLUSION

In this article, we considered a microgrid control problem to be solved over a peer-to-peer network of agents. Each agent represents a unit of the microgrid and must cooperate with the other units in order to solve the problem without a centralized coordinator. We used a challenging stochastic mixed-integer microgrid model and proposed a distributed algorithm to solve the problem, for which we provided theoretical guarantees on the constraint violation. Numerical computations on a synthesized problem using GANs show the validity of the proposed approach.

APPENDIX

A. Review of Primal Decomposition

Consider a network of N agents indexed by $\mathbb{I} = \{1, \dots, N\}$ that aim to solve a linear program of the form

$$\begin{aligned} \min_{x_1, \dots, x_N} \quad & \sum_{i=1}^N c_i^\top x_i \\ \text{s.t.} \quad & x_i \in X_i \quad \forall i \in \mathbb{I} \\ & \sum_{i=1}^N A_i x_i \leq b \end{aligned} \quad (25)$$

where each $x_i \in \mathbb{R}^{n_i}$ is the i th optimization variable, $c_i \in \mathbb{R}^{n_i}$ is the i th cost vector, $X_i \subset \mathbb{R}^{n_i}$ is the i th polyhedral constraint set, and $A_i \in \mathbb{R}^{S \times n_i}$ is a matrix for the i th contribution to the coupling constraint $\sum_{i=1}^N A_i x_i \leq b \in \mathbb{R}^S$. Problem (25) enjoys the constraint-coupled structure [3] and can be recast into a master-subproblem architecture by using the so-called *primal decomposition* technique [29]. The right-hand side vector b of the coupling constraint is interpreted as a given (limited) resource to be shared among the network agents. Thus, local allocation vectors $y_i \in \mathbb{R}^S$ for all i are introduced such that $\sum_{i=1}^N y_i = b$. To determine the allocations, a *master problem* is introduced

$$\begin{aligned} \min_{y_1, \dots, y_N} \quad & \sum_{i=1}^N p_i(y_i) \\ \text{s.t.} \quad & \sum_{i=1}^N y_i = b \\ & y_i \in Y_i \quad \forall i \in \mathbb{I} \end{aligned} \quad (26)$$

where, for each $i \in \mathbb{I}$, the function $p_i : \mathbb{R}^S \rightarrow \mathbb{R}$ is defined as the optimal cost of the i th (linear programming) *subproblem*

$$\begin{aligned} p_i(y_i) &= \min_{x_i} c_i^\top x_i \\ \text{s.t. } & A_i x_i \leq y_i \\ & x_i \in X_i. \end{aligned} \quad (27)$$

In problem (26), the new constraint set $Y_i \subseteq \mathbb{R}^S$ is the set of y_i for which problem (27) is feasible, i.e., such that there exists $x_i \in X_i$ satisfying the local *allocation constraint* $A_i x_i \leq y_i$. Assuming that problem (25) is feasible and X_i are compact sets, if (y_1^*, \dots, y_N^*) is an optimal solution of (26), and for all i , x_i^* is optimal for (27) (with $y_i = y_i^*$), then (x_1^*, \dots, x_N^*) is an optimal solution of the original problem (25) (see [29, Lemma 1]).

B. Proof of Theorem 1

By the optimality of $(x_i^\infty, \eta_i^\infty)$ for problem (24), it holds

$$c_i^\top x_i^\infty + d^\top \eta_i^\infty \leq c_i^\top x_i + d^\top \eta_i \quad (28)$$

for all $x_i \in X_i$ and $\eta_i \geq 0$ such that $A_i x_i^\infty \leq y_i^{\text{LP}} + \eta_i$. One vector satisfying such condition is $(x_i^{\text{L}}, \eta_i^{\text{L}})$ optimal solution of

$$\begin{aligned} \min_{x_i, \eta_i} & c_i^\top x_i + d^\top \eta_i \\ \text{s.t. } & 0 \leq \eta_i \leq M \mathbf{1}, \quad x_i \in X_i \\ & H_i x_i \leq \ell_i + \eta_i. \end{aligned} \quad (29)$$

Indeed, it holds $H_i x_i^{\text{L}} \leq \ell_i + \eta_i^{\text{L}} \leq y_i^{\text{LP}} + \eta_i^{\text{L}}$, where the first inequality is by construction, and the second one follows by the discussion above on ℓ_i . Thus, by using (28), we conclude that

$$d^\top \eta_i^\infty \leq c_i^\top (x_i^{\text{L}} - x_i^\infty) + d^\top \eta_i^{\text{L}}. \quad (30)$$

By explicitly writing the scalar product $d^\top \eta_i^\infty$ and by using the fact that $d, \eta \geq 0$, we obtain

$$d^\top \eta_i^\infty = \sum_{j=1}^{2RK} d_j \eta_{ij}^\infty \geq \underbrace{\left(\min_{j \in \{1, \dots, 2RK\}} d_j \right)}_{d^{\text{MIN}}} \sum_{j=1}^{2RK} \eta_{ij}^\infty.$$

Moreover, by using the fact that $\eta_{ij}^\infty \leq \sum_{k=1}^{2RK} \eta_{ik}^\infty$ for all k , we obtain

$$\eta_i^\infty \leq \frac{d^\top \eta_i^\infty}{d^{\text{MIN}}} \mathbf{1} \leq \frac{c_i^\top (x_i^{\text{L}} - x_i^\infty) + d^\top \eta_i^{\text{L}}}{d^{\text{MIN}}} \mathbf{1}.$$

Let us now compute an upper bound of the coupling constraint value, i.e.,

$$\begin{aligned} \sum_{i=1}^N H_i x_i^\infty - h &\leq \sum_{i=1}^N y_i^{\text{LP}} + \sum_{i \in I_Z} \eta_i^{\text{LP}} + \sum_{i \notin I_Z} \eta_i^\infty - h \\ &= \sum_{i \in I_Z} \eta_i^{\text{LP}} + \sum_{i \notin I_Z} \eta_i^\infty \end{aligned} \quad (31)$$

where we used the fact that, by Lemma 1, for $i \in I_Z$, it holds $H_i x_i^\infty \leq y_i^{\text{LP}} + \eta_i^{\text{LP}}$, while, for $i \notin I_Z$, it holds $H_i x_i^\infty \leq y_i^{\text{LP}} + \eta_i^\infty$. Thus, we, finally, obtain the bound

$$\sum_{i=1}^N H_i x_i^\infty - h \leq \sum_{i \in I_Z} \eta_i^{\text{LP}} + \sum_{i \notin I_Z} \frac{c_i^\top (x_i^{\text{L}} - x_i^\infty) + d^\top \eta_i^{\text{L}}}{d^{\text{MIN}}} \mathbf{1}$$

and the proof is complete. \square

REFERENCES

- [1] D. K. Molzahn *et al.*, "A survey of distributed optimization and control algorithms for electric power systems," *IEEE Trans. Smart Grid*, vol. 8, no. 6, pp. 2941–2962, Nov. 2017.
- [2] A. Nedić and J. Liu, "Distributed optimization for control," *Annu. Rev. Control Robot. Auton. Syst.*, vol. 1, pp. 77–103, May 2018.
- [3] G. Notarstefano, I. Notarnicola, and A. Camisa, "Distributed optimization for smart cyber-physical networks," *Found. Trends Syst. Control*, vol. 7, no. 3, pp. 253–383, 2020.
- [4] A. Ouammi, H. Dagdougui, L. Dessaint, and R. Sacile, "Coordinated model predictive-based power flows control in a cooperative network of smart microgrids," *IEEE Trans. Smart Grid*, vol. 6, no. 5, pp. 2233–2244, Sep. 2015.
- [5] S. R. Cominesi, M. Farina, L. Giulioni, B. Picasso, and R. Scattolini, "A two-layer stochastic model predictive control scheme for microgrids," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 1, pp. 1–13, Jan. 2018.
- [6] C. Le Floch, S. Bansal, C. J. Tomlin, S. J. Moura, and M. N. Zeilinger, "Plug-and-play model predictive control for load shaping and voltage control in smart grids," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2334–2344, May 2019.
- [7] A. Zakariazadeh, S. Jadid, and P. Siano, "Smart microgrid energy and reserve scheduling with demand response using stochastic optimization," *Int. J. Electr. Power Energy Syst.*, vol. 63, pp. 523–533, Dec. 2014.
- [8] T. A. Nguyen and M. L. Crow, "Stochastic optimization of renewable-based microgrid operation incorporating battery operating cost," *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 2289–2296, May 2016.
- [9] P. O. Kriett and M. Salani, "Optimal control of a residential microgrid," *Energy*, vol. 42, no. 1, pp. 321–330, 2012.
- [10] M. Marzband, M. Ghadimi, A. Sumper, and J. L. Domínguez-García, "Experimental validation of a real-time energy management system using multi-period gravitational search algorithm for microgrids in islanded mode," *Appl. Energy*, vol. 128, pp. 164–174, Sep. 2014.
- [11] E. Shirazi and S. Jadid, "Cost reduction and peak shaving through domestic load shifting and DERs," *Energy*, vol. 124, pp. 146–159, Apr. 2017.
- [12] A. Parisio, E. Rikos, and L. Glielmo, "A model predictive control approach to microgrid operation optimization," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 5, pp. 1813–1827, Sep. 2014.
- [13] A. Parisio, E. Rikos, and L. Glielmo, "Stochastic model predictive control for economic/environmental operation management of microgrids: An experimental case study," *J. Process Control*, vol. 43, pp. 24–37, Jul. 2016.
- [14] E. Espina, J. Llanos, C. Burgos-Mellado, R. Cardenas-Dobson, M. Martinez-Gomez, and D. Sáez, "Distributed control strategies for microgrids: An overview," *IEEE Access*, vol. 8, pp. 193412–193448, 2020.
- [15] F. Dörfler, S. Bolognani, J. W. Simpson-Porco, and S. Grammatico, "Distributed control and optimization for autonomous power grids," in *Proc. 18th Eur. Control Conf. (ECC)*, Jun. 2019, pp. 2436–2453.
- [16] S. Bolognani and S. Zampieri, "A distributed control strategy for reactive power compensation in smart microgrids," *IEEE Trans. Autom. Control*, vol. 58, no. 11, pp. 2818–2833, Nov. 2013.
- [17] V. Causevic, A. Falsone, D. Ioli, and M. Prandini, "Energy management in a multi-building set-up via distributed stochastic optimization," in *Proc. Annu. Amer. Control Conf. (ACC)*, Jun. 2018, pp. 5387–5392.
- [18] F. Belluschi, A. Falsone, D. Ioli, K. Margellos, S. Garatti, and M. Prandini, "Distributed optimization for structured programs and its application to energy management in a building district," *J. Process Control*, vol. 89, pp. 11–21, May 2020.
- [19] G. Cavraro, A. Bernstein, R. Carli, and S. Zampieri, "Distributed minimization of the power generation cost in prosumer-based distribution networks," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2020, pp. 2370–2375.
- [20] A. Falsone, K. Margellos, and M. Prandini, "A distributed iterative algorithm for multi-agent MILPs: Finite-time feasibility and performance characterization," *IEEE Control Syst. Lett.*, vol. 2, no. 4, pp. 563–568, Oct. 2018.
- [21] A. Camisa, I. Notarnicola, and G. Notarstefano, "Distributed primal decomposition for large-scale MILPs," *IEEE Trans. Autom. Control*, vol. 67, no. 1, pp. 413–420, Jan. 2022.
- [22] OPSP. (2020). *Open Power System Data Time Series*. Accessed: Oct. 6, 2020. [Online]. Available: https://data.open-power-system-data.org/time_series

- [23] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, Mar. 1999.
- [24] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [25] A. J. Kleywegt, A. Shapiro, and T. Homem-de Mello, "The sample average approximation method for stochastic discrete optimization," *SIAM J. Optim.*, vol. 12, no. 2, pp. 479–502, 2002.
- [26] A. Shapiro and A. Philpott, "A tutorial on stochastic programming," Tech. Rep., 2007. [Online]. Available: <https://cpn-us-w2.wpmucdn.com/sites.gatech.edu/dist4/1470/files/2021/03/TutorialSP.pdf>
- [27] F. Farina, A. Camisa, A. Testa, I. Notaricola, and G. Notarstefano, "DISROPT: A Python framework for distributed optimization," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 2666–2671, 2020.
- [28] I. J. Goodfellow *et al.*, "Generative adversarial networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014. [Online]. Available: <https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf>
- [29] G. J. Silverman, "Primal decomposition of mathematical programs by resource allocation: I—Basic theory and a direction-finding procedure," *Oper. Res.*, vol. 20, no. 1, pp. 58–74, Feb. 1972.



Andrea Camisa (Member, IEEE) received the Laurea degree (*summa cum laude*) in computer engineering from the University of Salento, Lecce, Italy, in 2017, the Licenza degree from the ISUFI Excellence School, Lecce, in 2018, and the Ph.D. degree in biomedical, electrical, and systems engineering from the University of Bologna, Bologna, Italy, in 2021.

He was a Visiting Student with the University of Stuttgart, Stuttgart, Germany, from 2017 to 2018. He is currently a Post-Doctoral Research Fellow and an Adjunct Professor with the University of Bologna. His research interests include reinforcement learning, and convex, distributed, and mixed-integer optimization.



Giuseppe Notarstefano (Member, IEEE) received the Laurea degree (*summa cum laude*) in electronics engineering from the University of Pisa, Pisa, Italy, in 2003, and the Ph.D. degree in automation and operation research from the University of Padua, Padua, Italy, in 2007.

He was an Assistant Professor, Ricercatore, from February 2007 to May 2016 and an Associate Professor from June 2016 to June 2018 with the University of Salento, Lecce, Italy. He has been a Visiting Scholar with the University of Stuttgart, Stuttgart, Germany, the University of California at Santa Barbara, Santa Barbara, CA, USA, and the University of Colorado Boulder, Boulder, CO, USA. He is currently a Professor with the Department of Electrical, Electronic, and Information Engineering, Alma Mater Studiorum University of Bologna, Bologna, Italy. His research interests include distributed optimization, cooperative control in complex networks, applied nonlinear optimal control, and trajectory optimization and maneuvering of aerial and car vehicles.

Dr. Notarstefano was a recipient of an ERC Starting Grant 2014. He also serves as an Associate Editor for IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY and IEEE CONTROL SYSTEMS LETTERS. He has been appointed as an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL. He is also part of the Conference Editorial Board of IEEE Control Systems Society and the European Control Association (EUCA).