Resilient and Privacy-Preserving Multi-Agent Optimization and Control of a Network of Battery Energy Storage Systems Under Attack

Mojtaba Kaheni, *Member, IEEE*, Elio Usai, *Senior Member, IEEE*, and Mauro Franceschelli, *Senior Member, IEEE*

***Abstract*— This paper deals with resilient and privacy-preserving control to optimize the daily operation costs of networked Battery Energy Storage Systems (BESS) in a multi-agent network vulnerable to various types of cyber-attacks. First, we formulate the optimization problem by defining the objective function and the local and coupling constraints. Next, we introduce a novel resilient decentralized control and optimization algorithm that can mitigate the effects of cyber-attacks, specifically false data injection attacks and hijacking, to enhance the network's resilience. The proposed method is based on filtering out outlier Lagrange multipliers in a suitable dual problem. Our proposed algorithm has two main advantages compared to the existing literature. Firstly, it can solve problems where the coupling constraint is not restricted to the average or a function of the average of decision variables. Secondly, our algorithm extends the well-known dual decomposition and Lagrange multiplier method to the decentralized control problem of BESSs. In the proposed algorithm presented in this paper, only the data relevant to the dual problem is exchanged among the agents. Noticing that the data of the dual problem does not contain any private information, mitigating privacy concerns associated with our proposed algorithm. We formally prove the convergence of our algorithm to a feasible and sub-optimal solution. Additionally, simulations demonstrate the effectiveness of our results.**

***Note to Practitioners*—Optimal coordinated control of BESSs increases the power system's reliability and reduces costs. With the expansion of the use of small-scale BESSs in household customers, it is possible to considerably increase the free capacity of power networks by optimally controlling these small BESSs. The methods published so far to solve such problems either share the private information of each BESS or are not resilient to failures or false data injection due to cyber-attacks. Therefore, these approaches are not favored in practical applications. Considering this practical motivation, in this paper, we present a decentralized algorithm to control a large set of BESSs in a platform vulnerable to various types of cyber-attacks without compromising privacy.**

***Index Terms*— Battery energy storage systems, multi-agent systems, resilient optimization.**

Manuscript received 16 June 2023; accepted 22 August 2023. Date of publication 7 September 2023; date of current version 16 October 2024. This article was recommended for publication by Associate Editor Q. Wei and Editor T. Nishi upon evaluation of the reviewers' comments. This work was supported in part by the Project "Network 4 Energy Sustainable Transition–NEST" funded under the National Recovery and Resilience Plan (NRRP), (Mission 4, Component 2, Investment 1.3–Call for tender No. 1561 of 11.10.2022), Ministero dell–UniversitÃe della Ricerca (MUR); in part by the European Union–NextGenerationEU, Code PE0000021, Concession Decree No. 1561 of 11.10.2022 adopted by MUR, under Grant CUP F53C22000770007; in part by Fondazione di Sardegna with Grant "Formal Methods and Technologies for the Future of Energy Systems," under Grant F72F20000350007; in part by the Knowledge Foundation (KKS) with Grant "Safe and Secure Adaptive Collaborative Systems (SACSys)," under Grant 20190021; and in part by the Swedish Agency for Innovation Systems (Vinnova) with Grant "GREENER: Intelligent Energy Management in Connected Construction Sites" under Grant 2019-05877. *(Corresponding author: Mauro Franceschelli.)*

Mojtaba Kaheni is with IDT, Mälardalen University, 721 23 Västerås, Sweden (e-mail: mojtaba.kaheni@mdu.se).

Elio Usai and Mauro Franceschelli are with DIEE, University of Cagliari, 09123 Cagliari, Italy (e-mail: elio.usai@unica.it; mauro.franceschelli@unica.it).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TASE.2023.3310328.

Digital Object Identifier 10.1109/TASE.2023.3310328

NOMENCLATURE

$A(k)$	Adjacency matrix.
$\Phi(k, s)$	Transition matrix.
$\lambda(k)$	Lagrange multipliers vector.
Q	Capacity of BESS.
ζ	State of charge (SoC).
p_b	Charging power of BESS.
p_b^l	Locally consumed discharging power of BESS.
p_b^s	Discharging power of BESS sold to the grid.
η	Round trip efficiency of BESS.
p_g	Total generated power in residential load.
p_g^l	Locally consumed generated power.
p_g^s	Power locally generated sold to the gird.
p_b^{max}	Max. allowable charging power of BESS.
p_b^{min}	Max. allowable discharging power of BESS.
Δt	Sampling time.
q	Total number of time samples.
ζ^{max}	Maximum allowable SoC of BESS.
ζ^{min}	Minimum allowable SoC of BESS.
m^1	Slope of the power limit on discharge.
m^2	Slope of the power limit on charge.
b^1	Intercept of the power limit on discharge.
b^2	Intercept of the power limit on charge.
p_l	Local power consumption of residential load.
p_l^{max}	Maximum allowable power transfer to the grid.
L^{max}	Max. allowable power transfer at the substation.
L	Predicted load of the substation.
w	Electrical energy price.
F	Maximum number of adversarial agents.
$\Lambda(x, \lambda)$	Lagrangian function.
$\varphi(\lambda)$	Dual function.
$P_X[.]$	Projection on X.
$\|.\|$	Set cardinality.
$\lfloor . \rfloor$	Floor function.
$(.)^T$	Matrix transpose.
$\Vert . \Vert$	2-norm (Euclidean norm).

I. Introduction

THE increased penetration of renewable energy sources in the energy market and the objective to decarbonize energy production to promote the transition to a green economy make widespread use of energy storage systems a necessity. Furthermore, domestic solar panels and battery energy storage benefit from high incentives, reducing the cost of adoption.

A large number of BESSs can reach the power level required for direct participation in the electricity market, according to the concept of Virtual Power Plants (VPP) [1], [2]. The optimal control of charge and discharge power of the set of BESSs to achieve maximum profit is one of the fundamental measures of VPP. Various methods to achieve this goal using centralized algorithms have been proposed in the literature so far [3], [4], [5], [6], [7], [8]. However, from a practical point of view, centralized methods have drawbacks as follows:

- The presence of large number of BESSs in a VPP leads to an increase in the number of decision variables in the optimization problem. Solving optimization problems with a substantial number of decision variables is a time-consuming process.
- All local information, including constraints and state of charges, needs to be transmitted to a central unit, which can potentially raise privacy concerns.

In the last decade, decentralized and distributed optimization has been introduced as a suitable solution to address the aforementioned drawbacks [9], [10], [11], [12]. Numerous articles from various perspectives have tackled power system-related issues using distributed optimization algorithms [13], [14], [15], [16], [17]. However, one concern when utilizing networked systems is the potential for cyber-attacks. Within a BESS network, these attacks can disrupt the system's optimal performance and even violate the defined constraints essential for maintaining electrical network stability. Therefore, in addition to developing network security measures to reduce the likelihood of cyber-attacks, research and development of resilient algorithms to counteract such attackers is deemed essential.

Resilient consensus-based distributed optimization is capable of handling false data injected by cyber-attacks. An early discussion on adversarial distributed optimization is presented in [18]. In this study, the authors explore the detrimental effects of misbehaving agents and propose an optimization algorithm that converges to a suboptimal solution within a defined interval. This interval prevents arbitrary alterations by the misbehaving agents. In the context of dealing with cyber-attacks, a similar method addressing deception attacks is developed in [19]. Furthermore, the authors in [20] enhance the accuracy of the solution by imposing additional restrictions on the communication network. Additionally, an algorithm based on gradient tracking is presented in [21] to tackle the issue, while the authors in [22] introduce an algorithm specifically designed for constrained problems in the shared memory architecture.

Resilient decentralized control and optimization of BESSs pose an *optimal resource allocation* problem. In resource allocation problems, each agent seeks to determine its optimal level of resource consumption while ensuring that common resource constraints are satisfied. In contrast, optimization problems involve all agents solving for a shared parameter.

To the best of our knowledge, the topic of resilient resource allocation has been addressed in a limited number of papers, including [23], [24], and [25]. In [23] and [24], the authors propose decentralized primal-dual optimization frameworks that incorporate a trusted coordinator connecting all agents. By utilizing the algorithm presented in [23] and [24], the computational burden is effectively distributed among different agents. On the other hand, in [25], agents can cooperate within an undirected, fixed graph without requiring a central coordinator. However, a notable drawback of these algorithms lies in the limitations imposed on the coupling constraint. For example, the coupling constraint is restricted to only being the average of the decision variables (as seen in [25]), or a function based on the average of the decision variables (as observed in [23] and [24]). In Section III, we elaborate on how the coupling constraint of the BESS planning problem, deviates from the setting proposed in prior works such as [23], [24], and [25]. Thus, our primary objective in this article is to present a novel resilient decentralized resource allocation algorithm capable of effectively handling our specific case study.

Furthermore, in the existing resilience resource allocation algorithms proposed so far, the communication of primal variable data among agents is necessary. However, in many applications, such as the control of BESSs that we focus on in this paper, preserving the privacy of primal variable information is crucial. To address this concern and prevent the sharing of information related to the primal problem among BESSs, we employ the method of Lagrange multipliers and dual decomposition in our approach. This strategy has been successfully utilized to solve optimization problems where privacy is of utmost importance [26], [27], [28]. Building upon the well-established Lagrange multipliers method and dual decomposition, we propose a novel resilient resource allocation algorithm for decentralized control and optimization of BESSs. Our algorithm exclusively utilizes dual variables data, thus eliminating the aforementioned privacy issues. A comparison of the resilient resource allocation algorithms in the literature with our proposed algorithm in this article is given in Table I.

A. Statement of Contributions

- Our paper proposes a novel algorithm addressing the planning and control of BESS network by introducing a more comprehensive coupling constraint setting, surpassing existing approaches in flexibility and applicability.
- The proposed algorithm preserves privacy of the local data of each BESS.
- We formally prove that the proposed algorithm is resilient against a number F of misbehaving BESS compromised

TABLE I

COMPARISON AGAINST OTHER RESILIENT RESOURCE ALLOCATION APPROACHES

	Turan et al. [23] and Xu et al. [24]	Wang et al. [25]	Algorithm 1
Network topology	Star graph	Undirected, fixed graph	Shared memory
Exchange of data related to	Primal problem	Primal problem	Dual problem
Coupling constraint structure	$g\left(\frac{1}{n}\sum_{i=1}^{n} x_i\right) \leq \mathbf{0}$	$\frac{1}{n}\sum_{i=1}^{n} x_i \leq s$	$\sum_{i=1}^{n} g_i(x_i) \leq \mathbf{0}$
Required assumptions	Convexity Compactness Strong duality $\nabla g(.)$ is L-Lipschitz	Convexity Compactness Strong duality $\nabla f(.)$ is L-Lipschitz	Convexity Compactness Strong duality $g(.)$ is L-Lipschitz

by false data injection attacks that compromise up to $F \leq \lfloor \frac{n-1}{4} \rfloor$ BESS, where n is the total number of BESSs connected to the network.

- We provide numerical simulations to corroborate our theoretical results.

B. Organization

The remainder of this paper is organized as follows. Following this introduction, Section II revisits fundamental preliminaries in multi-agent systems and resilient distributed optimization. Section III presents the problem statements and introduces the associated constraints. In Section IV, we propose our resilient decentralized control and optimization algorithm. The simulation results are discussed in Section V. Finally, Section VI concludes this article. A comprehensive nomenclature table is provided to explain the main notations used in this paper.

II. PRELIMINARIES

A. Network Modeling

In this paper, we assume that BESSs are connected to shared virtual memory. We consider that each BESS can read the memory areas of all others but can only write on its own, which is equivalent to allowing all the BESSs to communicate with each other and exchange data. Therefore, throughout the remainder of this article, we refer to each BESS as an *agent,* and we model the network of BESSs as *complete graphs*, denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Here $\mathcal{V} = \{1, 2, \ldots, n\}$ represents the set of agents, and $\mathcal{E} = \{\mathcal{V} \times \mathcal{V}\}$ represents the set of edges. We assume that each edge is associated with a weight, making $\mathcal{G}$ a *weighted graph*. The adjacency matrix of a weighted graph is denoted as $A = \left[a_{ij}\right] \in \mathbb{R}^{n\times n}$ where a_{ij} represents the weight associated with the information received by agent i from agent j. A weighted graph $\mathcal{G}(k)$ is time-varying if the weights $a_{ij}(k)$ are time-varying.

B. Helpful Lemmas and Definitions

In this subsection, we recall several helpful lemmas and definitions that we will use in this article.

Let $\Phi(k, s)$ represent the *transition matrix* defined as

$$\Phi(k, s) = A(k)A(k-1)\ldots A(s+1)A(s). \tag{1}$$

Lemma 1 describes the conditions for convergence in terms of connectivity and adjacency matrix weights.

Lemma 1 ([29]): Consider a network of agents with graph $\mathcal{G}(k)$. Assume that there exists a scalar $\tau \in (0, 1)$, such that $\forall i \in \mathcal{V}$, it holds $a_{ii}(k) \geq \tau$, and for all $i \neq j$, it holds either $a_{ij}(k) = 0$ or $a_{ij}(k) \geq \tau$. If $\mathcal{G}(k)$ is rooted and the respective adjacency matrix $A(k)$ is row stochastic, $\forall k$, then there exist two positive scalars $B > 0$ and $\xi \in (0, 1)$ and a stochastic vector $\phi(s) = [\phi_1(s), \phi_2(s), \ldots, \phi_n(s)]^T$ such that $\lim_{k\to\infty} \Phi(k, s) = \mathbf{1}_n \phi(s)^T$ and

$$\left\| [\Phi(k, s)]_{i,j} - \phi_j(s) \right\| \leq B\xi^{k-s}.$$

An essential step in the method of Lagrange multipliers is to project the Lagrangian multipliers into right hand plane. The following Lemma, establishes a relation between the projection error vector and the feasible directions of the convex set X at the projection vector.

Lemma 2 ([30]): Let $X \subset \mathbb{R}^n$ be a nonempty closed convex set, we have for any $x \in \mathbb{R}^n$ and $y \in X$

$$\| P_X[x] - y\|^2 \leq \|x - y\|^2 - \|P_X[x] - x\|^2.$$

Let $N_{in}^i(k)$ represent the in-neighbors of agent i. A subset $S \subset \mathcal{V}$ of agents is said to be *r-reachable* if there exists a agent $i \in S$ such that $\left|N_{in}^i \backslash S\right| \geq r$. In what follows, the *r-robust* graphs are formally defined.

Definition 1 ([18]): $\mathcal{G}$ is said to be r-robust if for all pairs of disjoint nonempty subsets, $S_1, S_2 \in \mathcal{V}$, at least one of S_1 or S_2 is r-reachable.

From Definition 1, we find that a complete graph with n agents is $\lfloor \frac{n+1}{2} \rfloor$-robust where $\lfloor . \rfloor$ represents the floor function. The following result is also helpful in our convergence analysis.

Lemma 3 ([31]): Suppose $\mathcal{G}$ is r-robust and $\mathcal{G}'$ be a graph obtained by removing $r-1$ or fewer incoming edges from each agent in $\mathcal{G}$. Then $\mathcal{G}'$ is rooted.

Another helpful Lemma that we use in this paper is about the infinite sum of products of the components of two sequences.

Lemma 4 ([30]): Let $0 < \beta < 1$ and assume $\{\gamma_k\}$ be a positive scalar sequence. If $\lim_{k\to\infty} \gamma_k = 0$. Then

$$\lim_{k\to\infty} \sum_{l=0}^{k} \beta^{k-l}\gamma_l = 0.$$

III. PROBLEM STATEMENT

As previously discussed, we assume that the BESSs are distributed across various locations within the power network, and they can share their computational results with each other at each time step through shared memory. For the purpose of analysis, we partition the total load of high to medium voltage transformers into two distinct components: the load associated with customers who have installed BESSs, and the load corresponding to the energy consumption of other residential customers.

We consider that the network consists of n independent BESSs, each can be modeled as [3]

$$Q_i \frac{\partial \zeta_i}{\partial t} = \eta_i p_{b_i} + p^l_{b_i} + p^s_{b_i}, \tag{2}$$

where $i = 1, \dots, n$ denotes different BESSs in the network, Q_i and ζ_i represent the capacity and the state of charge (SoC) of each BESS, respectively. The charging power is denoted by p_{b_i}. The power during a discharge is divided into two parts denoted as p^l_{bi} and p^s_{bi}. p^l_{bi} represents the power locally consumed during a discharge and p^s_{bi} is the power sold to the grid during a discharge. Throughout this paper, we quantify the charging power of BESSs by positive values, i.e. $p_{bi} \geq 0$, while the discharging power is quantified by negative values, i.e. $p^l_{bi}, p^s_{bi} \leq 0$. Finally, η_i represents the round trip efficiency.

Furthermore, we assume that a renewable generator (RG) may have been installed alongside the BESS. The RG's generated power can also be divided into two parts as follows

$$p_{g_i} = p^l_{g_i} + p^s_{g_i}, \tag{3}$$

where p_{g_i} stands for the total power generated by RG, $p^l_{g_i}$ and $p^s_{g_i}$ represent local consumption of the generated power (includes power consumption to charge the BESS) and power which is sold to the grid, respectively. We quantify p_{g_i}, $p^l_{g_i}$ and $p^s_{g_i}$ with negative values in this article.

Our objective in this paper is to propose a resilient algorithm to determine a sub-optimal control for the network of BESSs in 24 hours receding horizon time window. We denote the sampling time by Δt, thus the total number of samples is equal to $q = \frac{24}{\Delta t}$. Our goal is to find a resilient sub-optimal time-varying reference for $p_{b_i}, p^l_{b_i}, p^s_{b_i}, p^l_{g_i}, p^s_{g_i} \in \mathbb{R}^{q\times 1}$ in an adversarial environment, i.e. some agents misbehave and share false information with others due to a failure or a cyber-attack.

Each BESS is subject to local constraints and the network of BESSs is constraint coupled as discussed in the next subsections.

A. Local Constraints

1) Decision Variable Bounds: The decision variables $p_{b_i}, p^l_{b_i}, p^s_{b_i}, p^l_{g_i}$ and $p^s_{g_i}$ in each BESS are subject to the next bounds

$$\begin{aligned} \mathbf{0} \leq p_{b_i} &\leq {p_b}^{max}_i, \\ {p_b}^{min}_i \leq p^l_{b_i} &\leq \mathbf{0}, \\ {p_b}^{min}_i \leq p^s_{b_i} &\leq \mathbf{0}, \\ p_{g_i} \leq p^l_{g_i} &\leq \mathbf{0}, \\ p_{g_i} \leq p^s_{g_i} &\leq \mathbf{0}, \end{aligned} \tag{4}$$

where $p_{g_i} \in \mathbb{R}^{q\times 1}_{-}$ stands for estimated renewable power generation. Throughout this paper, we consider that the estimation of p_{g_i} is satisfactorily precise. The maximum allowable charge and discharge power of each BESS is defined by ${p_b}^{max}_i \in \mathbb{R}^{q\times 1}_{+}$ and ${p_b}^{min}_i \in \mathbb{R}^{q\times 1}_{-}$, respectively. The following constraints are derived by the definition of decision variables

$$\begin{aligned} {p_b}^{min}_i &\leq p^l_{b_i} + p^s_{b_i}, \\ p_{g_i} &= p^l_{g_i} + p^s_{g_i}. \end{aligned} \tag{5}$$

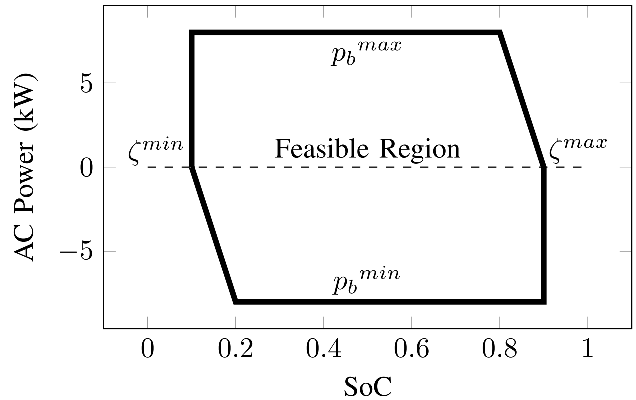

Fig. 1. Feasible region for ac power described by the kinetic battery model.

2) Bound on the State of Charge (SoC): The SoC of each BESSs must remain in the bound stated by their manufacturing companies.

$$\zeta^{min}_i \leq \zeta_i \leq \zeta^{max}_i, \tag{6}$$

where $\zeta^{max}_i \in \mathbb{R}^{(q+1)\times 1}_{+}$ and $\zeta^{min}_i \in \mathbb{R}^{(q+1)\times 1}_{+}$ represent the maximum and minimum allowable SoC, respectively. The dimensions of ζ^{min}_i,ζ_i and ζ^{max}_i vectors in (6) have been chosen $q+1$ to be appropriate for (7), and in fact the initial charge value has been added to ζ_i as its first index. It is worth noting that ζ_i is implicitly related to decision variables by

$$Q_i \mathbf{D} \zeta_i = \eta_i p_{b_i} + p^l_{b_i} + p^s_{b_i}, \tag{7}$$

where $\zeta_i \in \mathbb{R}^{(q+1)\times 1}_{+}$; and the matrix $\mathbf{D} \in \mathbb{R}^{q\times(q+1)}$ is defined as

$$\mathbf{D} = \frac{1}{\Delta t} \begin{bmatrix} -1 & 1 & 0 & . & . & 0 \\ 0 & -1 & 1 & 0 & . & 0 \\ & & . & . & & \\ & & & . & & \\ 0 & & & 0 & -1 & 1 \end{bmatrix}_{q\times(q+1)},$$

to approximate the derivative operator in (2).

3) Kinetic Battery Model: The relation between SoC and the power limits is usually represented by the kinetic battery model [3]. The feasible region for AC power described by the kinetic battery model is depicted in Fig. 1 which is actually the intersection of charge, discharge and SoC bounds with a region delimited by two lines,

$$m^1_i \zeta_i + b^1_i \leq p_{b_i} + p^l_{b_i} + p^s_{b_i} \leq m^2_i \zeta_i + b^2_i, \tag{8}$$

where m^1_i and b^1_i are the slope and intercept of the power limit on discharge, respectively, and m^2_i and b^2_i are the slope and intercept of the power limit on charge, respectively.

4) Local Power Limits: In residential loads, the power transfer between customers and the grid is limited due to the installed infrastructure. This constraint is expressed as:

$$\begin{aligned} p_{b_i} + p_{l_i} + p^l_{b_i} + p^l_{g_i} &\leq p_{l_i}^{max}, \\ -p^s_{b_i} - p^s_{g_i} &\leq p_{l_i}^{max}. \end{aligned} \tag{9}$$

where $p_{l_i} \in \mathbb{R}_+^{q\times 1}$ represent the local power consumption and $p_{l_i}^{max} \in \mathbb{R}_+^{q\times 1}$ is the upper bound of the residential load power transfer with the grid.

5) Constraints on Power Flow: We denoted with separate variables the power used to charge the BESS p_{b_i}, the power consumed by the local load p_{l_i}, the power output of the BESS during a discharge $p^l_{b_i}$ which is consumed locally, the local power generation consumed locally $p^l_{g_i}$.

Since the sum of the local power generation and the output power of the BESS consumed locally cannot exceed the actual local load consumption, it follows that:

$$p_{b_i} + p_{l_i} + p^l_{b_i} + p^l_{g_i} \geq \mathbf{0}. \tag{10}$$

B. Coupling Constraints

A large number of BESSs increases the potential short-term load variations of residential customers, posing a potential threat to the stability of the power distribution infrastructure. Specifically, to prevent overloads in high voltage substations, the following coupling constraints need to be satisfied:

$$-L^{max} - L \leq \sum_{i=1}^{n} (p_{b_i} + p^l_{b_i} + p^s_{b_i} + p^l_{g_i} + p^s_{g_i}) \leq L^{max} - L, \tag{11}$$

where $L^{max} \in \mathbb{R}_+^{q\times 1}$ is the maximum allowable power transfer at the substation and $L \in \mathbb{R}_+^{q\times 1}$ represent the predicted load of the substation, excluding the contributions from batteries and renewable generations. It is assumed throughout this paper that the predicted load L can be precisely estimated.

C. Objective Function

We define the decision variables vector of each BESS as

$$x_i = \left[p_{b_i}^T, p^{l\,T}_{b_i}, p^{s\,T}_{b_i}, p^{l\,T}_{g_i}, p^{s\,T}_{g_i} \right]^T. \tag{12}$$

Our goal is to minimize the operation cost of the network of BESSs which can be defined as

$$\begin{aligned} \min_{(x_i \in X_i)_{i=1}^n} \; & \sum_{i=1}^{n} f_i(x_i), \\ s.t. \; & \sum_{i=1}^{n} g_i(x_i) \leq \mathbf{0}, \end{aligned} \tag{13}$$

where

$$f_i(x_i) = w^T \left(p_{b_i} + p^l_{b_i} + p^s_{b_i} + p^l_{g_i} + p^s_{g_i} \right), \tag{14}$$

and $w \in \mathbb{R}^{q\times 1}$ represent the electrical energy price in different time intervals. There are many possible selections of local $g_i(x_i)$ in (13) by which $\sum_{i=1}^{n} g_i(x_i) \leq \mathbf{0}$ implies the coupling constraint in (11). For instance, we can assume that each agent contributes to the coupling constraint $\sum_{i=1}^{n} g_i(x_i) \leq \mathbf{0}$ via function $g_i : \mathbb{R}^{5q} \to \mathbb{R}^{2q}$, where

$$\begin{aligned} g_i(x_i) &= \left[g_i^1(x_i)^T, g_i^2(x_i)^T \right]^T, \\ g_i^1(x_i) &= p_{b_i} + p^l_{b_i} + p^s_{b_i} + p^l_{g_i} + p^s_{g_i} - \frac{L^{max} - L}{n}, \\ g_i^2(x_i) &= -p_{b_i} - p^l_{b_i} - p^s_{b_i} - p^l_{g_i} - p^s_{g_i} - \frac{L^{max} + L}{n}. \end{aligned} \tag{15}$$

We further assume that some agents may share incorrect data in the shared memory due to cyber-attacks or failures. In such situations, it is possible for the operation schedule obtained by solving a distributed resource allocation algorithm to significantly deviate from the optimal solution, thereby posing a potential threat to the stability of the network. Therefore, we seek an algorithm that can ensure acceptable performance even in adversarial environments.

Remark 1: Table I displays the coupling constraint settings found in existing literature on resilient decentralized resource allocation algorithms. The first setting, introduced in [23] and [24], is represented as $g\left(\frac{1}{n}\sum_{i=1}^n x_i\right) \leq \mathbf{0}$. Another setting, presented in [25], is expressed as $\frac{1}{n}\sum_{i=1}^n x_i \leq s$. It is important to note that neither of these settings aligns with (15), which is $\sum_{i=1}^n g_i(x_i) \leq \mathbf{0}$. ■

IV. Decentralized Resilient Control and Optimization of BESSs

The optimization problem given in (13) is commonly referred to as a *distributed resource allocation problem*. Each agent i has its own vector of decision variables denoted as x_i, a local constraint set denoted as X_i, and an objective function denoted as $f_i(.) : \mathbb{R}^{5q} \to \mathbb{R}$. Additionally, each agent i contributes to the coupling constraint $\sum_{i=1}^{n} g_i(x_i) \leq \mathbf{0}$ through $g_i(.) : \mathbb{R}^{5q} \to \mathbb{R}^{2q}$. We make the following assumptions regarding the resource allocation problem (13) and the communication network.

Assumption 1 (Slater's Condition): There exists a vector $\tilde{x} = \left[\tilde{x}_1 \ldots \tilde{x}_n\right] \in relint(X)$, where $relint(X)$ is the relative interior of the set X, such that $\sum_{i=1}^{n} g_i\left(\tilde{x}_i\right) \leq \mathbf{0}$ for those components of $\sum_{i=1}^{n} g_i(x_i)$ that are linear in x, while $\sum_{i=1}^{n} g_i\left(\tilde{x}_i\right) < \mathbf{0}$ for all other components.

Assumption 2: $c(k)$ is a non-increasing sequence of positive reals such that

$$c(k) > 0, \quad \sum_{k=0}^{\infty} c(k) = \infty, \quad \sum_{k=0}^{\infty} c^2(k) < \infty, \tag{16}$$

In this section, we present a decentralized resilient control and optimization algorithm for BESSs. The algorithm is designed to be *resilient against up to F adversarial agents*, meaning it can find a suboptimal solution to the optimization and control problem without being influenced by up to F unknown and undetected adversarial agents in the network who send false information to their neighbors. To mathematically explain this statement, let us consider the optimization problem (13) used to find an optimal reference for BESSs' SoC. We define $x^* \in X_1 \times X_2 \times \cdots \times X_n$ as the optimizer of (13), and $\lambda^* \in \mathbb{R}^{2q}$ as its corresponding dual pair. Furthermore, throughout this paper, we use the notation $\mathcal{V}_a$ to represent the set of adversarial agents, and $\mathcal{V}_r$ to represent the set of regular agents with nominal behavior. The sets $\mathcal{V}_r$ and $\mathcal{V}_a$ are thus disjoint and $\mathcal{V} = \mathcal{V}_a \cup \mathcal{V}_r$.

It is important to ensure that the majority of agents in the BESSs network are regular agents. This constraint is reasonable because in a network with a large number of adversarial agents, the resilient algorithm may fail to accurately detect the correct pattern and mistakenly identify regular agents as agents with abnormal behavior. To address this, we impose a bound on the number of adversarial agents. Specifically, we consider that the number of adversarial agents is bounded as follows:

$$2F + 1 \leq \lfloor \frac{n+1}{2} \rfloor, \tag{17}$$

where $\lfloor . \rfloor$ stands for floor function.

Algorithm 1 represents our proposed decentralized resilient control and optimization for BESSs. In this algorithm, for each index m ranging from 1 to $2q$, the algorithm isolates F agents with higher values of $\lambda_i^m(k)$ and F agents with lower values of $\lambda_i^m(k)$. This isolation process results in the reshaping of the complete graph of the BESSs network into $2q$ sequences of graphs denoted as $\mathcal{G}^m(k)$. Each $\mathcal{G}^m(k)$ consists of a complete graph with $n - 2F$ agents belonging to the set $\mathcal{V}_n^m(k)$, while $\mathcal{V}_h^m(k) \cup \mathcal{V}_l^m(k)$ contains $2F$ agents that only receive information and do not affect other agents.

From constraint (17), we observe that the reshaped complete graph of BESSs, $\mathcal{G}(k)$, becomes $2F+1$-robust. In Algorithm 1, each agent discards $2F$ of its incoming edges, which results in a rooted graph for each of the $2q$ sequences of graphs, $\mathcal{G}^m(k)$, as guaranteed by Lemma 3.

Consider the Lagrangian function as

$$\Lambda(x, \lambda) = \sum_{i=1}^{n} \Lambda_i(x_i, \lambda) = \sum_{i=1}^{n} \left\{ f_i(x_i) + \lambda^T g_i(x_i) \right\}, \tag{18}$$

where $x = \left[x_1^T, x_2^T, \ldots, x_n^T\right]^T \in X_1 \times X_2 \times \cdots \times X_n$ and $\lambda \in \mathbb{R}_+^{2q}$ is the Lagrange multipliers vector. Therefore, the dual function is

$$\varphi(\lambda) = \min_{x \in X} \{\Lambda(x, \lambda)\}. \tag{19}$$

Since the objective and constraints functions are separable in (18), (19) can be written as

$$\varphi(\lambda) = \sum_{i=1}^{n} \varphi_i(\lambda) = \sum_{i=1}^{n} \min_{x_i \in X_i} \Lambda_i(x_i, \lambda). \tag{20}$$

Therefore, the dual of (13) can be expressed as

$$\max_{\lambda \in \mathbb{R}_+^{2q}} \min_{x \in X} \Lambda(x, \lambda) = \max_{\lambda \in \mathbb{R}_+^{2q}} \sum_{i=1}^{n} \varphi_i(\lambda). \tag{21}$$

Theorem 1 states the main result of this article. We will prove Theorem 1 after bringing the necessary lemmas and discussions.

Algorithm 1 Decentralized Resilient Control of BESSs

Initialization
$k = 0$
$F =$ Maximum number of adversarial agents
$\hat{x}_i(0) \in X_i, \forall i \in \mathcal{V}$
$\lambda_i(0) \in \mathbb{R}_+^{2q}, \forall i \in \mathcal{V}$
while stopping criterion is not satisfied **do**
  Clustering
  Each agent gathers $\lambda_i(k), \forall i \in \mathcal{V}$
  for $m \in \{1, \ldots, 2q\}$ **do**
    Each agent builds the next sets of neighbors:
    $\mathcal{V}_h^m(k) = \{F \ agents \ with \ highest \ \lambda^m(k)\}$,
    $\mathcal{V}_l^m(k) = \{F \ agents \ with \ lowest \ \lambda^m(k)\}$,
    $\mathcal{V}_n^m(k) = \{Other \ agents\}$.
    Weights assignment
    Agent i assigns weights to its edges according to: $(\beta = n - 2F)$
    if $i \in \mathcal{V}_n^m(k)$ **then**
$$a_{ij}^m(k) = \begin{cases} \frac{1}{\beta} & \text{if} \quad j \in \mathcal{V}_n^m(k) \\ 0 & \text{otherwise.} \end{cases}$$
    else if $i \in \mathcal{V}_h^m(k) \cup \mathcal{V}_l^m(k)$ **then**
      if $i \in \mathcal{V}_h^m(k)$ **then**
$$i_r^m(k) = \arg\max_{j \in \mathcal{V}_n^m(k)} \lambda_j^m(k)$$
      else
$$i_r^m(k) = \arg\min_{j \in \mathcal{V}_n^m(k)} \lambda_j^m(k)$$
      end if
$$a_{ij}^m(k) = \begin{cases} \frac{1}{\beta} & \text{if} \quad j \in \left\{\mathcal{V}_n^m(k) \backslash i_r^m(k)\right\} \\ \frac{1}{\beta} & \text{if} \quad i = j \\ 0 & \text{otherwise.} \end{cases}$$
    end if
    State Update
$$l_i^m(k) = \sum_{j=1}^{n} a_{ij}^m(k) \lambda_j^m(k)$$
  end for
$$x_i(k+1) \in \arg\min_{x_i \in X_i} f_i(x_i) + l_i(k)^T g_i(x_i)$$
$$\lambda_i(k+1) = \arg\max_{\lambda_i \geq 0} \left\{ g_i(x_i(k+1))^T \lambda_i - \frac{1}{2c(k)} \|\lambda_i - l_i(k)\|^2 \right\}$$
$$\hat{x}_i(k+1) = \hat{x}_i(k) + \frac{c(k)}{\sum_{r=0}^{k} c(r)} \left(x_i(k+1) - \hat{x}_i(k) \right)$$
  $k \leftarrow k + 1$
end while

Theorem 1: Consider that Assumptions 1 and 2 hold and the complete graph representing communication between agents is $2F + 1$-robust. Consider that $P(\mathcal{V}_r)$ represents the powerset of $\mathcal{V}_r$. Let us define $\mathcal{V}_\Omega = \{x \in P(\mathcal{V}_r) \mid |x| = n - 2F\}$, i.e. $\mathcal{V}_\Omega$ is the set that contains all subsets of $\mathcal{V}_r$ of cardinality $n - 2F$. If we denote the converging value of the dual pair by executing Algorithm 1 by $\bar{\lambda} \in \mathbb{R}_+^{2q}$, then for all indices of $\bar{\lambda}$, that we denote by $\bar{\lambda}^m$, $m \in \{1, \ldots, 2q\}$,

$$\lambda_{min}^m \leq \bar{\lambda}^m \leq \lambda_{max}^m,$$

where λ^m_{min} and λ^m_{max} are defined by

$$\lambda^m_{min} = \min_{\Omega_j \in \mathcal{V}_\Omega} \left\{ \arg\max_{\lambda \in \mathbb{R}^{2q}_+} \sum_{i \in \Omega_j} \varphi^m_i(\lambda) \right\},$$

$$\lambda^m_{max} = \max_{\Omega_j \in \mathcal{V}_\Omega} \left\{ \arg\max_{\lambda \in \mathbb{R}^{2q}_+} \sum_{i \in \Omega_j} \varphi^m_i(\lambda) \right\},$$

and $\varphi^m_i(\lambda)$ represents the m^{th} index of $\varphi_i(\lambda)$, ■

To establish the effectiveness of Algorithm 1, it is crucial to demonstrate that the regular agents within the network achieve consensus on the Lagrange multiplier vector independently of any potential influence from adversarial agents. The following lemma presents a formal proof supporting this claim.

Lemma 5: If Assumptions 1 and 2 hold, and the communication network of BESSs is $2F+1$-robust, the Lagrange multipliers vector of all regular agents executing Algorithm 1 converge to a constant vector $\bar{\lambda}$, despite the presence of up to F adversarial agents. In addition, $\bar{\lambda}$ is independent of the adversarial agents' values.

$$\lim_{k\to\infty} \left\| \lambda_i(k) - \bar{\lambda} \right\| = 0, \forall i \in \mathcal{V}_r.$$

Proof: The maximization in the State Update phase of Algorithm 1 can be explicitly solved, leading to

$$\lambda_i(k+1) = P_{\mathbb{R}^{2q}_+}\left[l_i(k) + c(k) g_i(x_i(k+1))\right]. \tag{22}$$

We can rewrite (22) in the following form

$$\lambda_i(k+1) = l_i(k) + c(k) g_i(x_i(k+1)) + e_i(k), \tag{23}$$

where

$$\begin{aligned} e_i(k) = {} & P_{\mathbb{R}^{2q}_+}\left[l_i(k) + c(k) g_i(x_i(k+1))\right] \\ & - (l_i(k) + c(k) g_i(x_i(k+1))). \end{aligned} \tag{24}$$

We first consider the case where there are no adversaries in the network. From the State Update phase of Algorithm 1, we notice

$$l^m_i(k) = \sum_{j=1}^{n} a^m_{ij}(k) \lambda^m_j(k). \tag{25}$$

The Weights Assignment phase implies that for all $i \in \mathcal{V}$ and $m \in \{1, \ldots, 2q\}$, $\sum_{j=1}^{n} a^m_{ij}(k) = 1$. Therefore, $l^m_i(k)$ in (25) is a convex combination of $\lambda^m_i(k)$ where $i \in \mathcal{V}$. Since $\lambda^m_i(k) \in \mathbb{R}_+$, $l^m_i(k) \in \mathbb{R}_+$, which yields $l_i(k) \in \mathbb{R}^{2q}_+$.

If we replace $x = l_i(k) + c(k) g_i(x_i(k+1))$ and $y = l_i(k)$ in Lemma 2 we obtain

$$\begin{aligned} & \|\lambda_i(k+1) - l_i(k)\|^2 \\ & \le \|l_i(k) + c(k) g_i(x_i(k+1)) - l_i(k)\|^2 \\ & \quad - \|\lambda_i(k+1) - (l_i(k) + c(k) g_i(x_i(k+1)))\|^2 \\ & \le c(k)^2 \|g_i(x_i(k+1))\|^2 - \|e_i(k)\|^2, \end{aligned} \tag{26}$$

therefore,

$$\|e_i(k)\| \le c(k) \|g_i(x_i(k+1))\|. \tag{27}$$

According to the constraints defined in Section III, X_i is compact. Furthermore, (15) implies that $g(.)$ is L-Lipschitz in the compact set, X_i, therefore there exists a constant $\mathcal{L}$ such that

$$\|g_i(x_i(k+1))\| < \mathcal{L}, \tag{28}$$

for all $i \in \mathcal{V}$. Thus, considering (16)

$$\lim_{k\to\infty} c(k) \|g_i(x_i(k+1))\| = \lim_{k\to\infty} \|e_i(k)\| = 0. \tag{29}$$

For each element of $\lambda^m_i(k)$, where $m \in \{1, \ldots, 2q\}$, there exists a sequence of information exchange matrices $A^m(k) = \left[a^m_{ij}(k)\right]$. Let us denote the corresponding transition matrix of $\lambda^m_i(k)$ as

$$\Phi^m(k, s) = A^m(k) A^m(k-1) \ldots A^m(s+1) A^m(s). \tag{30}$$

From (23) and (25), $\lambda^m_i(k+1)$ is equal to

$$\lambda^m_i(k+1) = \sum_{j=1}^{n} a^m_{ij}(k) \lambda^m_j(k) + c(k) g^m_i(x_i(k+1)) + e^m_i(k). \tag{31}$$

Using the transition matrix defined in (30), we can rewrite (31) for all i, m and for all $k > s$ as follows

$$\begin{aligned} \lambda^m_i(k+1) = {} & \sum_{j=1}^{n} \left[\Phi^m(k, s)\right]_{ij} \lambda^m_j(s) \\ & + \sum_{r=s}^{k-1} \sum_{j=1}^{n} \left[\Phi^m(k, r+1)\right]_{ij} c(r) g^m_j(x_i(r+1)) \\ & + c(k) g^m_j(x_i(k+1)) \\ & + \sum_{r=s}^{k-1} \sum_{j=1}^{n} \left[\Phi^m(k, r+1)\right]_{ij} e^m_j(r) + e^m_i(k). \end{aligned} \tag{32}$$

Therefore, considering (29), we have

$$\begin{aligned} & \lim_{k\to\infty} \left\{ \lambda^m_{i_1}(k+1) - \lambda^m_{i_2}(k+1) \right\} \\ & = \lim_{k\to\infty} \sum_{j=1}^{n} \left(\left[\Phi^m(k, s)\right]_{i_1 j} - \left[\Phi^m(k, s)\right]_{i_2 j} \right) \lambda^m_j(s) \\ & \quad + \lim_{k\to\infty} \sum_{r=s}^{k-1} \sum_{j=1}^{n} \left(\left[\Phi^m(k, r+1)\right]_{i_1 j} - \left[\Phi^m(k, r+1)\right]_{i_2 j} \right) \\ & \quad \ldots c(r) g^m_j(x_i(r+1)) \\ & \quad + \lim_{k\to\infty} \sum_{r=s}^{k-1} \sum_{j=1}^{n} \left(\left[\Phi^m(k, r+1)\right]_{i_1 j} - \left[\Phi^m(k, r+1)\right]_{i_2 j} \right) e^m_j(r). \end{aligned} \tag{33}$$

According to Algorithm 1, each agent discards, at most, the information received from $2F$ agents in its neighborhood to construct $A^m(k)$. Considering (17) and Lemma 3, the resulting graph is rooted. Therefore, based on Lemma 1, we have

$$\lim_{k\to\infty} \left(\left[\Phi^m(k, s)\right]_{i_1 j} - \left[\Phi^m(k, s)\right]_{i_2 j} \right) = 0. \tag{34}$$

Thus, the first term of (33) tends to zero. In the second and third terms of (33), Lemma 1 leads to

$$\begin{aligned} -B \xi^{k-(r+1)} & \le \sum_{j=1}^{n} \left(\left[\Phi^m(k, r+1)\right]_{i_1, j} - \left[\Phi^m(k, r+1)\right]_{i_2, j} \right) \\ & \le B \xi^{k-(r+1)}, \end{aligned} \tag{35}$$

with $B > 0$ and $0 < \xi < 1$. According to (28), we can write the following inequality

$$-2n\mathcal{L}c(r) \leq c(r)\sum_{j=1}^{n} g_j^m(x_i(r+1)) + \sum_{j=1}^{n} e_j^m(r) \leq 2n\mathcal{L}c(r). \tag{36}$$

Thus, from Lemma 1 and (36), we can conclude that the overall term of the second and third terms in (33) is bounded within the interval

$$[-2n\mathcal{L}B\sum_{r=s}^{k-1}\xi^{k-(r+1)}c(r), 2n\mathcal{L}B\sum_{r=s}^{k-1}\xi^{k-(r+1)}c(r)]. \tag{37}$$

In view of Assumption 2 and Lemma 4, both extremes of the interval tend to zero. By applying the same reasoning for any component $m \in 1,\ldots,2q$, it holds that

$$\lim_{k\to\infty}\{\lambda_{i_1}(k) - \lambda_{i_2}(k)\} = 0, \tag{38}$$

In case there are adversarial agents in the network but they are isolated by Algorithm 1, considering that these agents have no effect on the updated values of regular agents, the situation is the same and regular agents reach a consensus.

If in some sub-graphs $\mathcal{G}^m(k)$, where $m \in \{1,\ldots,2q\}$, there are $K^m(k)$ adversarial agents in the network, where $1 \leq K^m(k) \leq F$, that are not filtered, there must be $K^m(k)$ regular agents that are filtered with a lower state value and also $K^m(k)$ regular agents that are filtered with a higher state value compared to the remaining adversarial agents within the ball of β agents with normal states. Therefore, the state of each adversarial agent can be expressed as a linear combination of two regular agents.

$$\{\lambda_a\}_i^m(k) = \sigma_i^m(k)\{\lambda_h\}_i^m(k) + \big(1-\sigma_i^m(k)\big)\{\lambda_l\}_i^m(k), \tag{39}$$

for $i = 1,\ldots,K^m(k)$ and $0 < \sigma_i^m(k) < 1$ where $\{\lambda_h\}_i^m(k)$ and $\{\lambda_l\}_i^m(k)$ represent regular agents that belong to $\mathcal{V}_h^m(k)$ and $\mathcal{V}_l^m(k)$, respectively, and $\{\lambda_a\}_i^m(k)$ are the adversarial agents. Therefore, $\mathcal{G}^m(k)$ is equivalent to a subgraph in which two filtered regular agents replace each of the remaining adversarial agents in the ball of β agents. It can be considered that instead of the adversarial agent state, the states of these two regular agents are communicated to their neighbors. So in this situation, $\mathcal{G}^m(k)$ is mathematically equivalent to a subgraph in which the adversarial agents are filtered and have no effect on the network. Thus, in any case, (38) holds true by executing Algorithm 1.

At this point, we will prove that the consensus value is a constant vector, denoted as $\bar{\lambda}$. Recall that the adjacency matrix of the mathematically equivalent graph achieved by (39), where adversarial agents are filtered out, is row stochastic. Therefore, from (38) and the definition of $l_i^m(k)$ in the State Update phase of Algorithm 1, for all $i \in \mathcal{V}_r$, it follows that

$$\begin{aligned}\lim_{k\to\infty} l_i^m(k) &= \lim_{k\to\infty}\sum_{j=1}^{n} a_{ij}^m(k)\lambda_j^m(k)\\ &= \lim_{k\to\infty}\sum_{j\in\mathcal{V}_r} a_{ij}^m(k)\lambda_j^m(k) = \lim_{k\to\infty}\lambda_i^m(k).\end{aligned} \tag{40}$$

On the other hand, the update rule in (23) leads to

$$\lim_{k\to\infty}\lambda_i^m(k+1) = \lim_{k\to\infty}\{l_i^m(k) + c(k)g_i^m(x_i(k+1)) + e_i^m(k)\}. \tag{41}$$

Since in view of (29) it holds that $\lim_{k\to\infty} c(k)g_i^m(x_i(k+1)) = \lim_{k\to\infty} e_i^m(k) = 0$, by combining (40) and (41), one obtains

$$\lim_{k\to\infty}\lambda_i^m(k+1) = \lim_{k\to\infty}\lambda_i^m(k), \tag{42}$$

which holds $\forall m$, $m \in \{1,\ldots,2q\}$ and shows that consensus vector of regular agents is constant, proving the desired result. ■

Next, we prove that by executing Algorithm 1, there exist time steps denoted as T^m, such that for all $k \geq T^m$ and for all $i \in \mathcal{V}_r$, $\lambda_i^m(k)$ are clustered with respect to $g_i^m(.)$.

Lemma 6: Under Assumptions 1 and 2, by executing Algorithm 1, there exist a time step $T^m < \infty$, where for all regular agents such as $i_1 \in \mathcal{V}_l^m(k)$, $i_2 \in \mathcal{V}_n^m(k)$, $i_3 \in \mathcal{V}_h^m(k)$ and $m \in \{1,\ldots,2q\}$, $\forall k > T^m$ it holds

$$g_{i_1}^m\big(x_{i_1}(k+1)\big) < g_{i_2}^m\big(x_{i_2}(k+1)\big) < g_{i_3}^m\big(x_{i_3}(k+1)\big).$$

Proof: First notice that from State Update phase of Algorithm 1, $\forall k > 0$ one gets

$$l_{i_1}^m(k) - l_{i_2}^m(k) = \frac{1}{\beta}\Big(\lambda_{i_1}^m(k) - \lambda_{i_r^m(k)}^m(k)\Big), \tag{43}$$

On the other hand, from (24) we have

$$\begin{aligned}e_i^m(k) = {} & P_{R_+}\big[l_i^m(k) + c(k)g_i^m(x_i(k+1))\big]\\ & - \big(l_i^m(k) + c(k)g_i^m(x_i(k+1))\big).\end{aligned} \tag{44}$$

Thus, we notice that

$$\begin{aligned}&l_i^m(k) + c(k)g_i^m(x_i(k+1)) \geq 0 \Rightarrow e_i^m(k) = 0,\\ &l_i^m(k) + c(k)g_i^m(x_i(k+1)) < 0 \Rightarrow e_i^m(k) > 0.\end{aligned} \tag{45}$$

Therefore, in any case $\forall k > 0$

$$e_i^m(k) \geq 0. \tag{46}$$

From (15), we notice that $g_i(\cdot)$ is continuous. By virtue of Lemma 5, we observe that $g_{i_1}^m\big(x_{i_1}(k+1)\big)$, $g_{i_2}^m\big(x_{i_2}(k+1)\big)$, and $g_{i_3}^m\big(x_{i_3}(k+1)\big)$ converge to different constant values according to the consensus value of states. Assume there exists a time step s such that the ordering of $g_{i_1}^m\big(x_{i_1}(k+1)\big)$, $g_{i_2}^m\big(x_{i_2}(k+1)\big)$, and $g_{i_3}^m\big(x_{i_3}(k+1)\big)$ does not change for all $k \geq s$. In other words, if $g_{i_1}^m\big(x_{i_1}(s+1)\big) < g_{i_2}^m\big(x_{i_2}(s+1)\big)$, then $g_{i_1}^m\big(x_{i_1}(k+1)\big) < g_{i_2}^m\big(x_{i_2}(k+1)\big)$ for all $k \geq s$, for all i_1 and i_2.

From (43), the difference $l_{i_2}^m(k-1) - l_{i_1}^m(k-1)$ can be written as:

$$l_{i_2}^m(k-1) - l_{i_1}^m(k-1) = \frac{1}{\beta}\Big(\lambda_{i_r^m(k-1)}^m(k-1) - \lambda_{i_1}^m(k-1)\Big), \tag{47}$$

Thus, by replacing (47) in (23) one gets

$$\begin{aligned}\lambda_{i_2}^m(k) - \lambda_{i_1}^m(k) = {} & \frac{1}{\beta}\Big(\lambda_{i_r^m(k-1)}^m(k-1) - \lambda_{i_1}^m(k-1)\Big)\\ & + c(k-1)\big(g_{i_2}^m\big(x_{i_2}(k)\big) - g_{i_1}^m\big(x_{i_1}(k)\big)\big)\\ & + e_{i_2}^m(k-1) - e_{i_1}^m(k-1).\end{aligned} \tag{48}$$

At this point, we suppose by way of contradiction that

$$\text{as} \quad k \to \infty \quad g_{i_1}^m\big(x_{i_1}(k+1)\big) > g_{i_2}^m\big(x_{i_2}(k+1)\big). \tag{49}$$

First notice that we observe from (45) that

$$e_{i_2}^m(k) \neq 0 \Rightarrow \lambda_{i_2}^m(k+1) = 0, \tag{50}$$

Therefore, $\lambda_{i_2}^m(k+1) \leq \lambda_{i_1}^m(k+1)$, which is a contradiction with the definition of $\mathcal{V}_l^m(k)$ and $\mathcal{V}_n^m(k)$. Thus, $e_{i_2}^m(k) = 0$, $\forall k > s$.

Let us define

$$\alpha^m(k) = c(k)\big(g_{i_2}^m\big(x_{i_2}(k+1)\big) - g_{i_1}^m\big(x_{i_1}(k+1)\big)\big) - e_{i_1}^m(k). \tag{51}$$

Thus, the difference $\lambda_{i_2}^m(k) - \lambda_{i_1}^m(k)$ in (48) can be written as:

$$\begin{aligned}\lambda_{i_2}^m(k) - \lambda_{i_1}^m(k) = \frac{1}{\beta}\Big(\lambda_{i_r^m(k-1)}^m(k-1) - \lambda_{i_1}^m(k-1)\Big)\\ + \alpha^m(k-1).\end{aligned} \tag{52}$$

From (46) and (49), we can deduce that for all $k > s$, $\alpha^m(k) < 0$. Therefore,

$$\lambda_{i_2}^m(k) - \lambda_{i_1}^m(k) < \frac{1}{\beta}\Big(\lambda_{i_r^m(k-1)}^m(k-1) - \lambda_{i_1}^m(k-1)\Big), \tag{53}$$

where $i_r^m(k-1) \in \mathcal{V}_n^m(k-1)$ by construction. We first analyze the case in which $i_r^m(k-1) \in \mathcal{V}_n^m(k-2)$ or $i_r^m(k-1) \in \mathcal{V}_h^m(k-2)$, i.e., the removed agent at time step $k-1$ was contained in $\mathcal{V}_n^m(k-2) \cup \mathcal{V}_h^m(k-2)$ at the previous time step. From the definition of $i_r^m(k)$ in Weights Assignment phase of Algorithm 1, we observe that in this case it holds $g_{i_2}^m\big(x_{i_1}(k-1)\big) > g_{i_r^m(k-1)}^m\big(x_{i_r^m(k-1)}(k-1)\big)$. Otherwise, we would obtain $\lambda_{i_2}^m(k-1) < \lambda_{i_r^m(k-1)}^m(k-1)$, which is a contradiction. Thus, similar to the procedure in (51) to (53) one can get

$$\begin{aligned}\lambda_{i_r^m(k-1)}^m(k-1) - \lambda_{i_1}^m(k-1)\\ < \frac{1}{\beta}\Big(\lambda_{i_r^m(k-2)}^m(k-2) - \lambda_{i_1}^m(k-2)\Big).\end{aligned} \tag{54}$$

Therefore, extending (53) to s, it holds

$$\lambda_{i_2}^m(k) - \lambda_{i_1}^m(k) < \frac{1}{\beta^{k-s}}\Big(\lambda_{i_r^m(s)}^m(s) - \lambda_{i_1}^m(s)\Big). \tag{55}$$

Otherwise if $i_r^m(k-1) \in \mathcal{V}_l^m(k-2)$, i.e., the agent $i_r^m(k-1) \in \mathcal{V}_n^m(k-1)$ was previously contained in $\mathcal{V}_l^m(k-2)$ at time step $k-2$. In this case, there must exist an agent $i_s^m(k-1)$ where $g_{i_r^m(k-1)}^m\big(x_{i_r^m(k-1)}(k-1)\big) > g_{i_s^m(k-1)}^m\big(x_{i_s^m(k-1)}(k-1)\big)$, which switched its position with $i_r^m(k-1)$, i.e., such that $i_s^m(k-1) \in \mathcal{V}_n^m(k-2) \cup \mathcal{V}_h^m(k-2)$ and $i_r^m(k-1) \in \mathcal{V}_l^m(k-1)$. This implies that the switching is a step towards sorting the agents with respect $g_i^m(.)$. We can then reinitialize $s = k$ and go forward again. Since the number of agents is bounded and all the replacements are towards sorting the agents, there exist finite time steps in which it holds $i_r^m(k) \in \mathcal{V}_l^m(k-1)$. Considering a time step s in which $i_r^m(k) \in \mathcal{V}_n^m(k-1) \cup \mathcal{V}_h^m(k-1)$ for all $k > s$, the same case as in the above is obtained. Therefore, since $\beta > 1$, from (55) one can obtain

$$\lim_{k\to\infty}\big(\lambda_{i_2}^m(k) - \lambda_{i_1}^m(k)\big) < 0, \tag{56}$$

which is a contradiction with the definition of $\mathcal{V}_l^m(k)$ and $\mathcal{V}_n^m(k)$. Applying a similar reasoning, one can reach a similar results for $i_2 \in \mathcal{V}_n^m(k)$ and $i_3 \in \mathcal{V}_h^m(k)$. This concludes the proof. ■

Now we are ready to discuss the resilient sub-optimality of Algorithm 1.

Proof of Theorem 1: First notice $\lambda_i(k+1)$ in Algorithm 1 can be explicitly solved, yields to

$$\lambda_i(k+1) = P_{\mathbb{R}_+^{2q}}\big[l_i(k) + c(k)g_i(x_i(k+1))\big]. \tag{57}$$

Reminding that

$$\varphi_i(\lambda) = \min_{x_i \in X_i}\big\{f_i(x_i) + \lambda^T g_i(x_i)\big\}, \tag{58}$$

From the application of Danskin's theorem [32], we can conclude that $g_i(x_i(k+1))$ is a subgradient of $\varphi_i(\lambda)$ evaluated at $\lambda = l_i(k)$. Moreover, since X_i is a compact set (as indicated in (4)), we can infer that $g_i(x_i(k+1))$ is bounded (as stated in (15)).

Based on Lemma 6, there exists a time step T^m where the regular agents do not swap their positions between the sets $\mathcal{V}_h^m(k)$, $\mathcal{V}_l^m(k)$, and $\mathcal{V}_n^m(k)$ for $k > T^m$. Considering $k > T^m$, we can define the sets Ω_h^m and Ω_l^m that contain β regular agents with the highest and lowest m^{th} coordinates of their Lagrange multiplier vectors λ_i^m, respectively. It is important to note that the agents in $\mathcal{V}_n^m(k)$ are sorted according to their gradients by the update rule. Hence, based on Lemma 6 and the fact that $\left|\mathcal{V}_l^m(k)\right| = \left|\mathcal{V}_h^m(k)\right| < \beta$, the sets Ω_h^m and Ω_l^m remain unchanged for all $k > T^m$. Additionally, we introduce the following auxiliary variables

$$\begin{aligned}\overline{M} &= \arg\max_{\lambda\in\mathbb{R}^+} \sum_{i\in\Omega_h^m} \varphi_i(\lambda)\\ \underline{m} &= \arg\max_{\lambda\in\mathbb{R}^+} \sum_{i\in\Omega_l^m} \varphi_i(\lambda)\end{aligned} \tag{59}$$

From Lemma 5, we can conclude that the regular agents reach consensus even in the presence of adversaries in the network. To prove the theorem, we need to show that the following inequality holds true for all components of the consensus value

$$\underline{m}^m \leq \bar{\lambda}^m \leq \overline{M}^m, \tag{60}$$

with $m = 1, \ldots, 2q$. Suppose by way of contradiction that $\bar{\lambda}^m = \overline{M}^m + \epsilon$, with $\epsilon > 0$. We define the following variables

$$M^m(k) = \sum_{i\in\Omega_h^m} \lambda_i^m(k), m^m(k) = \sum_{i\in\Omega_l^m} \lambda_i^m(k). \tag{61}$$

Since the number of adversaries is bounded by F, and recalling $\left|\Omega_h^m\right| = \beta = n - 2F$, there must exist at least F regular agents with smaller state values $\lambda_i^m(k)$ in the network that do not belong to Ω_h^m. This implies that $\mathcal{V}_l^m(k) \cap \Omega_h^m = \emptyset$ for all $k > T$. By a similar argument, we have $\mathcal{V}_h^m(k) \cap \Omega_l^m = \emptyset$ for all $k > T$. Thus, according to the update procedure in Algorithm 1, the following inequality holds true

$$\frac{1}{\beta}m^m(k) \leq l_i^m(k) \leq \frac{1}{\beta}M^m(k), \tag{62}$$

which according to (23), leads to

$$\lambda_i^m(k+1) \leq \frac{1}{\beta}M^m(k) + c(k)g_i^m(x_i(k+1)) + e_i^m(k). \quad (63)$$

Considering (63) and (61) for $M^m(k+1)$, it follows

$$M^m(k+1) \leq M^m(k) + c(k)\sum_{i\in\Omega_h^m} g_i^m(x_i(k+1)) + \sum_{i\in\Omega_h^m} e_i^m(k), \quad (64)$$

which can be generalized for $M(k+Z)$ as

$$M^m(k+Z) \leq M^m(k) + \sum_{k=k_0}^{Z-1}\left(c(k)\sum_{i\in\Omega_h^m} g_i^m(x_i(k+1)) + \sum_{i\in\Omega_h^m} e_i^m(k)\right). \quad (65)$$

Since we assumed $\bar{\lambda}^m = \overline{M}^m + \epsilon$, there exists a time step $k_0 > T^m$ such that the following inequality is verified for $k \geq k_0$ and $i \in \mathcal{V}_r$

$$\overline{M}^m + \frac{1}{2}\epsilon \leq \lambda_i^m(k) \leq \overline{M}^m + 2\epsilon, \quad (66)$$

which, summing for all $j \in \Omega_h^m$ and considering the definition in (61), leads to

$$\beta\left(\overline{M}^m + \frac{1}{2}\epsilon\right) \leq M^m(k) \leq \beta\left(\overline{M}^m + 2\epsilon\right). \quad (67)$$

In view of (59) and since the dual of a convex problem is concave, we observe that if it holds $\lambda_i^m(k) = \overline{M}^m$ for all $i \in \Omega_h^m$, then the sum of respective gradients are null, i.e., it holds $\sum_{i\in\Omega_h^m} g_i^m(x_i(k+1)) = 0$. Therefore, in order to fulfill (66) it must hold $\sum_{i\in\Omega_h^m} g_i^m(x_i(k+1)) < 0$ for $k > k_0$. At this point, recalling that $\sum_{k=k_0}^{\infty} c(k) = \infty$, and noticing that (66) is implicitly equivalent with $\sum_{i\in\Omega_h^m} e_i^m(k) = 0$, we obtain from (65) that, for a large enough Z, it holds $M^m(k_0+Z) < \beta(\overline{M}^m + \frac{1}{2}\epsilon)$, which is in contradiction with (67). Applying a similar reasoning, one can reach $\underline{m}^m \leq \bar{\lambda}^m$. This concludes the proof. ■

Theorem 1 demonstrates that the vector of Lagrange multipliers converges to a neighborhood, and the bounds of this neighborhood are independent of the behavior of the adversarial agents. Additionally, the primal vector converges to its corresponding neighborhood around the saddle point and equals $\varphi(\bar{\lambda})$. As a result, we achieve a suboptimal solution that remains unaffected by the presence of adversarial agents.

Remark 2: To ensure that the coupling constraints are satisfied, we account for the worst-case behavior of the $2F$ *filtered agents in the network. Specifically, we modify both* $g_1(x_i)$ *and* $g_2(x_i)$ *in (15) based on the maximum allowable power transfer of each agent, denoted as* $p_{l_i}^{max}$*. Let us define* P_M *as the maximum value of* $p_{l_i}^{max}$ *across all agents, i.e.,*

$$P_M = \max_{i\in\mathcal{V}} p_{l_i}^{max}.$$

In the worst-case scenario, the neglected agents can consume or inject a total power of P_M *into the network, thereby occupying the common resources. Therefore, we replace* L_{max} *with* $L_{max} - 2F \cdot P_M$ *in (15) to ensure that the coupling constraints are not violated.* ■

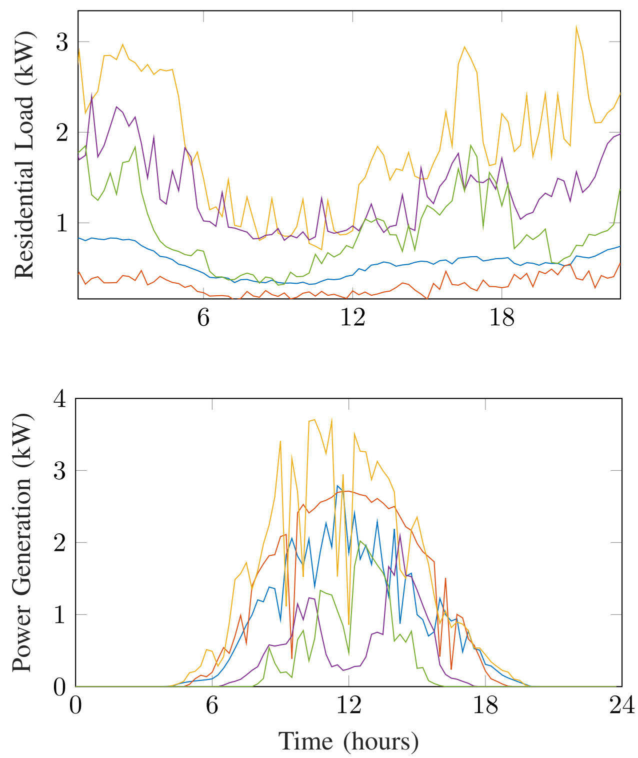

Fig. 2. Five different residential daily Load and renewable power generation types.

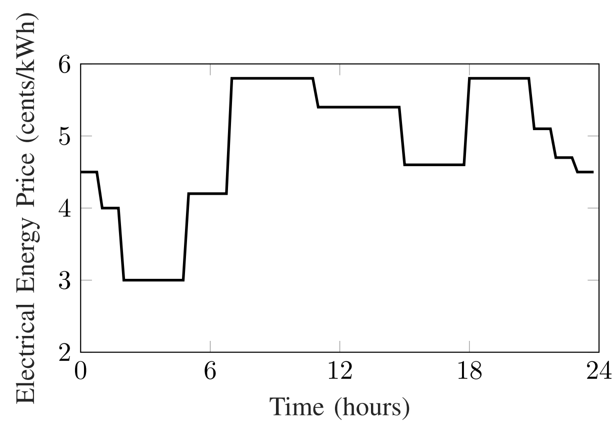

Fig. 3. Electrical energy tariff in a sample day.

V. Numerical Simulations

In this section, we evaluate the performance of Algorithm 1 and compare it with the non-resilient resource allocation algorithm introduced in [26]. We consider a network consisting of 100 BESSs. The data published in [33] and [34] is used to generate five different residential loads and five different renewable power generation types, respectively. The BESSs, residential loads, and renewable power generations are selected uniformly at random from the five different types presented in Table II and Fig. 2.

In this example, we assume $\Delta t = 15$ minutes, $p_{l_i}^{max} = 7$ kW for all BESSs, $L^{max} = 1500$ kW, and the electrical energy tariff for all BESSs is as depicted in Fig. 3. The assumed L and L^{max} are shown in Fig. 4. By installing batteries and implementing the method introduced in [26],

TABLE II
BATTERY TYPES PARAMETERS

Name	Symbol	Type 1	Type 2	Type 3	Type 4	Type 5
Energy Capacity	Q	25 kWh	20 kWh	22 kWh	17 kWh	15 kWh
Energy Efficiency	η	80%	70%	75%	85%	90%
Maximum Discharge Power	$pb^{\min}$	-10 kW	-8 kW	-9 kW	-7 kW	-5 kW
Maximum Charge Power	$pb^{\max}$	10 kW	8 kW	9 kW	7 kW	5 kW
Maximum SoC	ς^{max}	95%	90%	92%	88%	98%
Initial SoC	$\varsigma(0)$	30%	30%	30%	30%	30%
Minimum SoC	ς^{min}	20%	10%	15%	10%	5%
Kinetic model discharge slope	m^1			$pb^{\min}/10\%$		
Kinetic model discharge intercept	b^1			$-\varsigma^{\min} \times m^1$		
Kinetic model charge slope	m^2			$-pb^{\max}/5\%$		
Kinetic model charge intercept	b^2			$-\varsigma^{\max} \times m^2$		

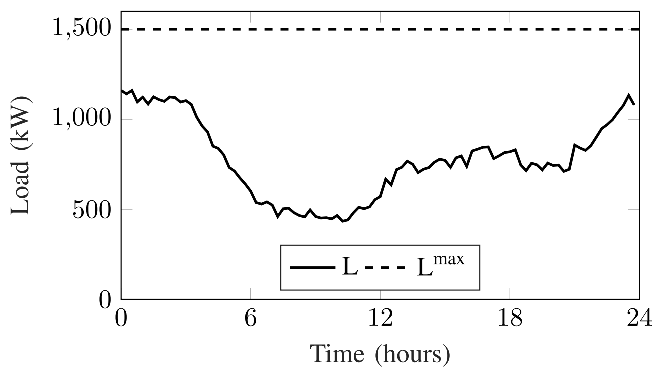

Fig. 4. The predicted load of substation by neglecting batteries, L, and its maximum allowable power transfer L^{max}.

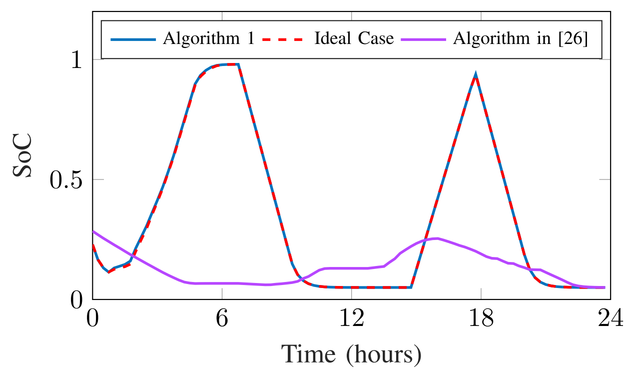

Fig. 5. Comparison of a randomly selected agent's SoC in a network consisting of 96 regular agents and 4 adversarial agents.

the daily energy cost of the network of 100 residential loads with renewable power generation reduces from 28.30 USD to -16.45 USD.

Although the algorithm introduced in [26] works perfectly under ideal conditions, it is highly sensitive to the presence of cyberattacks. As discussed in detail in [26], all agents in the network must agree on an identical Lagrange multiplier vector. At each step, all agents communicate their estimates of the Lagrange multiplier vector to each other. Therefore, if a few agents in a large network become adversarial, they can manipulate the consensus process by forcing the network to converge to their desired vector of Lagrange multipliers.

For instance, consider a network of 100 BESSs where 4 agents, denoted as $a_1, \ldots, a_4$, are adversarial. These adversarial agents consistently communicate a constant vector of Lagrange multipliers to all their out-neighbors in every step as follows

$$\lambda_{a_1}(k) = \lambda_{a_2}(k) = \lambda_{a_3}(k) = \lambda_{a_4}(k) = 1000 \times \mathbf{1}_{2q\times 1}. \quad (68)$$

The daily energy cost of the network of BESSs significantly deviates from the ideal case when implementing the algorithm proposed in [26]. The cost incurred by the network increases to 42.68 USD, which is even higher than the cost without the installation of BESSs. This highlights the vulnerability of the algorithm to adversarial agents and the negative impact they can have on the overall system performance.

In this example, in the absence of adversarial agents, the consensus value of the Lagrange multipliers vector is observed to be $\mathbf{0}_{2q\times 1}$. This indicates that the local optimal solutions of the agents do not violate the coupling constraints. However, when adversarial agents are added to the network with the values defined in (68), they induce the network that the coupling constraint is not met. As a result, the agents are forced to deviate from their optimal values (the values they would have achieved in the absence of coupling constraints) in order to satisfy the coupling constraint.

Similarly, in a network without the presence of adversarial agents, it is possible for the Lagrange multipliers vector to be greater than zero. However, the introduction of adversarial agents with the intention of reducing this value can lead to a situation where the agents' responses violate the coupling constraint. This highlights the need for the implementation of resilient algorithms that can effectively counter cyber-attacks and ensure the integrity of the system.

After implementing Algorithm 1 with $F = 4$ in a network under our proposed adversarial setting, the total operation cost of the network is -14.64 USD, which is close to the ideal case. This indicates that the algorithm effectively mitigates the impact of adversarial agents and achieves a cost-effective operation. Furthermore, in Fig. 5, the SoC of a randomly selected agent is plotted. It can be observed that by implementing Algorithm 1, the SoC closely resembles the ideal case, indicating that Algorithm 1 ensures proper battery management even in the presence of adversarial agents.

The maximum number of adversarial agents, denoted as F, is another effective parameter in Algorithm 1. As stated in Theorem 1, lower values of F can ensure better accuracy compared to higher values of F. In Fig. 6, the achieved

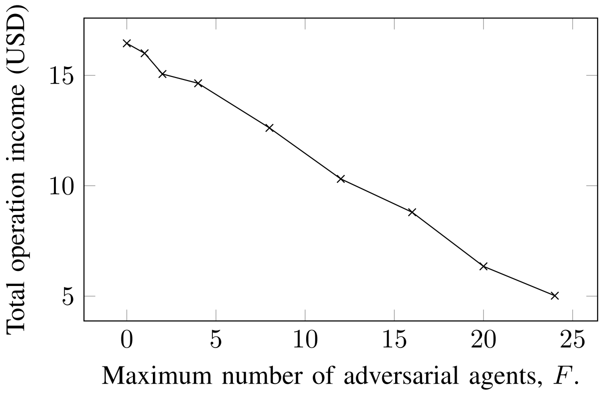

Fig. 6. The total operation income of the network achieved by implementing Algorithm 1 with various values of F. The income is negative of cost defined in (14).

total operation income of the network is illustrated when implementing Algorithm 1 with various values of F. In this case, we assume that adversarial agent i sends $\lambda_{a_i}(k) = 1000 \times \mathbf{1}_{2q\times 1}$ to its neighbors at each time step.

Remark 3: It must be noted that the actual number of adversarial agents is not known in advance and resiliency is not guaranteed if the number of actual adversaries exceeds F. The value of F should be selected by the user at the beginning of Algorithm 1. A proper value for F should satisfy two criteria. First, $F \leq \lfloor \frac{n-1}{4} \rfloor$, to guarantee that the resulting graph remains rooted. Next, it must be considered that small values of F threaten the resiliency of the network since the actual number of adversaries might exceed F. On the other hand, unnecessarily large selections of F reduce accuracy, as shown in Fig. 6. Therefore, a good design should find a trade-off between accuracy and resiliency. ■

Let us spend some words and discuss about the simulation time of Algorithm 1. The simulation time of our case study in MATLAB by using a laptop equipped with the Intel 10th generation core i7 processor was 935 seconds, which is equal to 15 minutes and 35 seconds. The simulation time almost linearly depends on the number of BESSs, therefore in our case study, the simulation process for each agent is around 9.35 seconds. In our simulation, since a single central processor accomplishes the necessary process of agents on behalf of them, the processing time has multiplied by the number of agents. Therefore, in practice, if each agent has its own processing unit, processing time will decrease significantly and it would be around 9.35 seconds. It must be noted that our code is not optimized and our programming language is not the best option if the goal is decreasing simulation time.

As discussed in Section I, we aim to propose an algorithm that works for residential BESSs networks which represents a network with a large number of agents. Therefore in this setting, the number of BESSs could be as many as the number of residential clients in a neighborhood or even larger scales, for instance, a city or a region. Thus, the number of BESS could be in the range of several thousands or even more. Since the computations in our proposed method are decentralized, our proposed algorithm is scalable, and the processing time is not dependent on the number of agents. Only some low time-consuming functions such as average, could take minor influences. The most time-consuming part, which is the optimization process at each time step, is independent of the number of agents. Scalability is one of the advantages we get from our proposed method compared to the centralized algorithms.

VI. Conclusion

In this paper, we presented a novel decentralized resilient control and optimization scheme for a network of BESSs. Our proposed scheme ensures the preservation of privacy of local data for each BESS while providing resilience against false data injection attacks. We have formally proven that our proposed decentralized control and optimization algorithm is resilient against a certain number F of misbehaving BESSs, where F satisfies the condition $F \leq \lfloor \frac{n-1}{4} \rfloor$, with n being the total number of BESSs in the network.

Numerical simulations have been conducted to validate the theoretical results and demonstrate the performance of the proposed method compared to a similar algorithm without embedded resiliency features. The simulations have shown the effectiveness of our approach in achieving cost-effective operation and maintaining battery State of Charge (SoC) close to the ideal case, even in the presence of adversarial agents.

Future research directions can focus on extending the proposed methods to more complex scenarios and relaxing the topological constraints on the communication graph. This would allow for the application of the resilient control and optimization scheme in larger and more diverse networks of BESSs, enhancing the applicability of the proposed approach.

References

[1] Z. Yi, Y. Xu, J. Zhou, W. Wu, and H. Sun, "Bi-level programming for optimal operation of an active distribution network with multiple virtual power plants," *IEEE Trans. Sustain. Energy*, vol. 11, no. 4, pp. 2855–2869, Oct. 2020.

[2] A. Baringo, L. Baringo, and J. M. Arroyo, "Day-ahead self-scheduling of a virtual power plant in energy and reserve electricity markets under uncertainty," *IEEE Trans. Power Syst.*, vol. 34, no. 3, pp. 1881–1894, May 2019.

[3] D. M. Rosewater, D. A. Copp, T. A. Nguyen, R. H. Byrne, and S. Santoso, "Battery energy storage models for optimal control," *IEEE Access*, vol. 7, pp. 178357–178391, 2019.

[4] R. H. Byrne, T. A. Nguyen, D. A. Copp, B. R. Chalamala, and I. Gyuk, "Energy management and optimization methods for grid energy storage systems," *IEEE Access*, vol. 6, pp. 13231–13260, 2018.

[5] L. H. Macedo, J. F. Franco, M. J. Rider, and R. Romero, "Optimal operation of distribution networks considering energy storage devices," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2825–2836, Nov. 2015.

[6] P. Malysz, S. Sirouspour, and A. Emadi, "An optimal energy storage control strategy for grid-connected microgrids," *IEEE Trans. Smart Grid*, vol. 5, no. 4, pp. 1785–1796, Jul. 2014.

[7] A. Raghavan, P. Maan, and A. K. B. Shenoy, "Optimization of day-ahead energy storage system scheduling in microgrid using genetic algorithm and particle swarm optimization," *IEEE Access*, vol. 8, pp. 173068–173078, 2020.

[8] S.-K. Kim, J.-Y. Kim, K.-H. Cho, and G. Byeon, "Optimal operation control for multiple BESSs of a large-scale customer under time-based pricing," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 803–816, Jan. 2018.

[9] T. Yang et al., "A survey of distributed optimization," *Annu. Rev. Control*, vol. 47, pp. 278–305, Jan. 2019.

[10] A. Nedić and J. Liu, "Distributed optimization for control," *Annu. Rev. Control Robot. Auton. Syst.*, vol. 1, pp. 77–103, May 2018.

[11] Y. Zhang and M. M. Zavlanos, "A consensus-based distributed augmented Lagrangian method," in *Proc. IEEE Conf. Decis. Control (CDC)*, Dec. 2018, pp. 1763–1768.

[12] A. Falsone, I. Notarnicola, G. Notarstefano, and M. Prandini, "Tracking-ADMM for distributed constraint-coupled optimization," *Automatica*, vol. 117, Jul. 2020, Art. no. 108962.

[13] L.-N. Liu and G.-H. Yang, "Distributed fixed-time optimal resource management for microgrids," *IEEE Trans. Autom. Sci. Eng.*, vol. 20, no. 1, pp. 404–412, Jan. 2023.

[14] D. Wu, T. Yang, A. A. Stoorvogel, and J. Stoustrup, "Distributed optimal coordination for distributed energy resources in power systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 2, pp. 414–424, Apr. 2017.

[15] M. Kaheni, E. Usai, and M. Franceschelli, "A distributed optimization and control framework for a network of constraint coupled residential BESSs," in *Proc. IEEE 17th Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2021, pp. 2202–2207.

[16] Y. Yang, G. Hu, and C. J. Spanos, "HVAC energy cost optimization for a multizone building via a decentralized approach," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 4, pp. 1950–1960, Oct. 2020.

[17] A. Cherukuri and J. Cortés, "Distributed generator coordination for initialization and anytime optimization in economic dispatch," *IEEE Trans. Control Netw. Syst.*, vol. 2, no. 3, pp. 226–237, Sep. 2015.

[18] S. Sundaram and B. Gharesifard, "Distributed optimization under adversarial nodes," *IEEE Trans. Autom. Control*, vol. 64, no. 3, pp. 1063–1076, Mar. 2019.

[19] W. Fu, Q. Ma, J. Qin, and Y. Kang, "Resilient consensus-based distributed optimization under deception attacks," *Int. J. Robust Nonlinear Control*, vol. 31, no. 6, pp. 1803–1816, Apr. 2021.

[20] L. Su and N. H. Vaidya, "Byzantine-resilient multiagent optimization," *IEEE Trans. Autom. Control*, vol. 66, no. 5, pp. 2227–2233, May 2021.

[21] N. Ravi, A. Scaglione, and A. Nedic, "A case of distributed optimization in adversarial environment," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 5252–5256.

[22] M. Kaheni, E. Usai, and M. Franceschelli, "Resilient constrained optimization in multi-agent systems with improved guarantee on approximation bounds," *IEEE Control Syst. Lett.*, vol. 6, pp. 2659–2664, 2022.

[23] B. Turan, C. A. Uribe, H.-T. Wai, and M. Alizadeh, "Resilient primal–dual optimization algorithms for distributed resource allocation," *IEEE Trans. Control Netw. Syst.*, vol. 8, no. 1, pp. 282–294, Mar. 2021.

[24] C. Xu, Q. Liu, and T. Huang, "Resilient penalty function method for distributed constrained optimization under Byzantine attack," *Inf. Sci.*, vol. 596, pp. 362–379, Jun. 2022.

[25] R. Wang, Y. Liu, and Q. Ling, "Byzantine-resilient resource allocation over decentralized networks," *IEEE Trans. Signal Process.*, vol. 70, pp. 4711–4726, 2022.

[26] A. Falsone, K. Margellos, S. Garatti, and M. Prandini, "Dual decomposition for multi-agent distributed optimization with coupling constraints," *Automatica*, vol. 84, pp. 149–158, Oct. 2017.

[27] A. Simonetto and H. Jamali-Rad, "Primal recovery from consensus-based dual decomposition for distributed convex optimization," *J. Optim. Theory Appl.*, vol. 168, no. 1, pp. 172–197, Jan. 2016.

[28] K. Tjell and R. Wisniewski, "Privacy preservation in distributed optimization via dual decomposition and ADMM," in *Proc. IEEE 58th Conf. Decis. Control (CDC)*, Dec. 2019, pp. 7203–7208.

[29] M. Cao, A. S. Morse, and B. D. O. Anderson, "Reaching a consensus in a dynamically changing environment: A graphical approach," *SIAM J. Control Optim.*, vol. 47, no. 2, pp. 575–600, Jan. 2008.

[30] A. Nedic, A. Ozdaglar, and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 922–938, Apr. 2010.

[31] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram, "Resilient asymptotic consensus in robust networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 4, pp. 766–781, Apr. 2013.

[32] D. P. Bertsekas, *Nonlinear Programming*. Nashua, NH, USA: Athena Scientific, 1999.

[33] C. J. Meinrenken et al., "MFRED, 10 second interval real and reactive power for groups of 390 U.S. apartments of varying size and vintage," *Sci. Data*, vol. 7, no. 1, p. 375, Nov. 2020.

[34] (2017). *Photovoltaic (PV) Solar Panel Energy Generation Data*. [Online]. Available: https://data.london.gov.uk/dataset/photovoltaic–pv–solar-panel-energy-generation-data

Mojtaba Kaheni (Member, IEEE) received the M.Sc. and Ph.D. degrees in control engineering from the Shahrood University of Technology, Shahrood, Iran, in 2011 and 2019, respectively. He is a Post-Doctoral Researcher with Akademin för Innovation, Design och Teknik (IDT), Mälardalen University, Västerås, Sweden. He was a Visiting Scholar with the University of Florence, Italy, from May 2017 to October 2017, and a Post-Doctoral Researcher with the University of Cagliari, Italy, from August 2020 to December 2022. His research interests include distributed optimization, multi-agent systems, and nonlinear control.

Elio Usai (Senior Member, IEEE) received the M.Sc. degree in electrical engineering from the University of Cagliari, Cagliari, Italy, in 1985. He was a process engineer and then the production manager of international companies. In September 1994, he joined the Department of Electrical and Electronic Engineering (DIEE), University of Cagliari, where he is currently a Professor of Automatic Control. He was a Coordinator of the Quality Team, University of Cagliari, from 2015 to 2021, where he is also a Rector's Delegate of the Quality of Processes and Services. He has been a leader of research projects on the control of uncertain systems and model-based fault detection. He has coauthored over 200 articles published in international journals and conference proceedings. His current interests are in output-feedback control, state estimation, and FDI via higher order sliding modes and multi-agent control systems. He was the General Co-Chairperson of the 2006 International Workshop on Variable Structure Systems. He has served as an Associate Editor for the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY. He is currently an Associate Editor of the *Asian Journal of Control*, the IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS: SYSTEMS, the *Journal of the Franklin Institute* and *Franklin Open*.

Mauro Franceschelli (Senior Member, IEEE) received the Laurea (cum laude) and Ph.D. degrees in electronic engineering from the University of Cagliari, Italy, in 2007 and 2011, respectively. He is an Associate Professor with the Department of Electrical and Electronic Engineering, University of Cagliari. He spent visiting periods with the Georgia Institute of Technology (GaTech), and the University of California at Santa Barbara (UCSB), USA. In 2013, he received a fellowship from the National Natural Science Foundation of China (NSFC), Xidian University, Xi'an, China. In 2015, he was an Assistant Professor (RTD-A) funded by the Italian Ministry of Education, University and Research (MIUR), under the 2014 call "Scientific Independence of Young Researchers" (SIR). His research interests include consensus problems, gossip algorithms, multi-agent systems, multi-robot systems, nonsmooth analysis, distributed optimization, and electric demand side management.

He has been a member of the Conference Editorial Board (CEB) for the IEEE Control Systems Society (CSS), since 2019. He is an Officer (Secretary) of the IEEE Italy Section Chapter, IEEE Control Systems Society. He has been serving as an Associate Editor for the IEEE Conference on Automation Science and Engineering (CASE), since 2015, the IEEE American Control Conference (ACC), since 2019, and IEEE Conference on Decision and Control, since 2020. He has been serving as an Associate Editor for the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, since 2021, and *Nonlinear Analysis: Hybrid Systems* (IFAC), since 2023.

Open Access funding provided by 'Università degli Studi di Cagliari' within the CRUI CARE Agreement