



HAL
open science

Decentralized optimal management of a large-scale EV fleet: optimality and computational complexity comparison between an Adaptive MAS and MILP

Sharyal Zafar, Anne Blavette, Guy Camilleri, Hamid Ben Ahmed, Jesse James Arthur Prince Agbodjan

► To cite this version:

Sharyal Zafar, Anne Blavette, Guy Camilleri, Hamid Ben Ahmed, Jesse James Arthur Prince Agbodjan. Decentralized optimal management of a large-scale EV fleet: optimality and computational complexity comparison between an Adaptive MAS and MILP. *International Journal of Electrical Power & Energy Systems*, 2022, 10.1016/j.ijepes.2022.108861 . hal-03891003

HAL Id: hal-03891003

<https://hal.science/hal-03891003>

Submitted on 8 Dec 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Decentralized optimal management of a large-scale EV fleet: optimality and computational complexity comparison between an Adaptive MAS and MILP

Sharyal Zafar^{a,*}, Anne Blavette^{a,b}, Guy Camilleri^c, Hamid Ben Ahmed^a, Jesse-James Prince Agbodjan^{a,b}

^aSATIE lab, ENS Rennes, Bruz 35170 France

^bCNRS, 3 Rue Michel Ange, Paris 75016, France

^cIRIT, Université de Toulouse, CNRS, Toulouse INP, UT3, Toulouse, France

Abstract

Increasing the penetration of variable and uncertain renewables and electric vehicles in power systems may give rise to problems (such as network congestion and commitment mismatches) if not controlled strategically. This demands control solutions in the form of energy management strategies for active distribution networks which would control the connected distributed energy resources and storage units in real-time to address the mentioned challenges. Centralized strategies may fail to serve this purpose for large-scale distribution networks due to their inherent shortcomings like vulnerability to single point of failures and large computing times. Unlike centralized approaches, decentralized control strategies show more potential. This paper presents one such solution, based on an adaptive multi-agent system, to control a large-scale distribution network in real-time. Its performance is compared with the results obtained with the corresponding centralized optimization problem, modeled as a mixed integer linear programming problem. Both the centralized version and the decentralized multi-agent version of the problem under consideration are presented and a case study is designed for the comparison. The comparison shows that the designed multi-agent system produces a near-optimal solution in real-time while the centralized optimization strategy struggles in terms of computational complexities for larger distribution networks.

Keywords: Active distribution networks; adaptive real-time control; electric vehicles; computational complexity; multi-agent system.

Nomenclature

$P_a(t)$	Bus a active power at time t
$P_{PV,a}(t)$	Bus a PV production at time t
$P_{load,a}(t)$	Bus a load active power at time t
$\tilde{P}_{EV,a}(t)$	Bus a planned EV active power at time t
$\Delta P_{EV,a}(t)$	Bus a change in planned EV active power at time t
$V_a(t)$	Bus a voltage at time t
$I_{ab}(t)$	Bus a to b electrical current at time t
$SoC_a(t)$	Bus a EV's state of charge at time t
$SoC_{a,min}$	Bus a EV's minimum state of charge at time t
$SoC_{a,max}$	Bus a EV's maximum state of charge at time t

* Corresponding author. Tel.: +33-661-322252.

E-mail address: sharyal.zafar@ens-rennes.fr

Preprint submitted to International Journal of Electrical Power & Energy Systems

$SoC_{a,min,depart}$	Bus a EV's desired state of charge at its departure time
$SoH_a(t)$	Bus a EV's state of health at time t
$E_{bat,a}$	Bus a EV's battery capacity
$P_{EV,charg,a}(t)$	Bus a EV's charging active power at time t
$P_{EV,discharg,a}(t)$	Bus a EV's discharging active power at time t
$P_{EV,a,max}$	Bus a EV's rated active power
P_{BRP}	Active power consumption/production in the BRP perimeter
$\tilde{P}(N)$	Active power commitment of the BRP during imbalance settlement period N
N	Ongoing BRP imbalance settlement period
$\tilde{P}_{PV}(t)$	Planned day-ahead active PV production at time t
$\tilde{P}_{load}(t)$	Planned day-ahead active load consumption at time t
$\tilde{P}_{EV}(t)$	Planned day-ahead active EV power at time t
$e_{PV}(t)$	Error in the predicted day-ahead PV forecast at time t
$e_{load}(t)$	Error in the predicted day-ahead load forecast at time t
$Cr_l(t)$	Criticality of the line agent at time t
$Cr_{i,l}(t)$	Incremental criticality of the line agent at time t
$Cr_b(t)$	Criticality of the bus agent at time t
$Cr_{i,b}(t)$	Incremental criticality of the bus agent at time t
$Cr_{BRP}(t)$	Criticality of the BRP agent at time t
$Cr_{EV}(t)$	Criticality of the EV agent at time t
$Cr_{ant}(t)$	Criticality of the antagonistic agent at time t
$I(t)$	Electrical current at time t
I_{max}	Electrical current rated value
$V(t)$	Voltage at time t
V_{max}	Voltage rated value
V_u	Voltage nominal value
$t_{a,depart}$	EV a departure time
tr_{period}	Time remaining before the end of the current BRP imbalance settlement period
K_l	Line agent's incremental criticality tuning parameter
K_b	Bus agent's incremental criticality tuning parameter
K_{BRP}	BRP agent's tuning parameter
h	Weighting function for the antagonist request
η_{charg}	Charging efficiency of the EV
$\eta_{discharg}$	Discharging efficiency of the EV
η_{PV}	Efficiency of the PV production
$Irr(t)$	Irradiance value at time t
A	Area of the PV panels

Introduction

The growth in adoption of distributed energy resources (DERs) and electrical vehicles (EVs) has opened the door to carbon-free energy systems but uncontrolled grid connections of such resources may give rise to issues such as voltage limit violations and electrical line congestions. The impact is not limited to only local levels, as due to imperfect forecasts, consumption/production mismatches can be observed on a global level. Different actors are playing different roles in the present energy markets and are responsible for managing these issues. The distribution

system operator (DSO) must ensure the reliable operation of the distribution network while the balance responsible party (BRP) is financially responsible for the energy mismatches between the planned and the real-time consumption/production. Each BRP has a set of loads and power sources in its perimeter, called the balance perimeter (BP). The BRP submits a day-ahead power consumption/production schedule to the transmission system operator (TSO). The BRP schedule is defined on a sub-hourly basis (soon to be harmonized to a quarter-hourly basis in Europe), known as the imbalance settlement period. The BRP is paid by the TSO if the

production is more during the day compared to the submitted schedule, and vice versa.

To reduce the cost of such imbalances, a BRP can utilize flexible entities (such as EVs) connected within its perimeter. But such control must not result in the violation of the constraints of other market actors, and/or should limit the degradation of other actors' objectives. These interactions of different market actors, at different levels, with possibly conflicting objectives, make the problem even harder. Grid reinforcement may solve the issues of local voltage/current constraints. However, it may be very expensive and time-consuming. Hence, flexible solutions may be preferred to tackle these challenges. Different approaches have been proposed in the literature which can be categorized as centralized and decentralized strategies. The major shortcomings of centralized approaches are their vulnerability to single point of failures, high communication overhead, dramatic growth of the solver time as the number of system variables and constraints are increased, and privacy concerns over the sharing of data to a third party. Decentralized approaches can tackle these drawbacks.

The multi-agent paradigm is particularly suitable to design decentralized systems. This paradigm is very flexible and can be easily adapted to the needs of decentralized applications. Multi-agent systems have been used in a wide range of fields and applications [1]. A multi-agent system is composed of several entities called agents. The definition of an agent often varies between researchers, fields, and applications. Russel and Norvig in [2] defined it as, "An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators." A more shared vision of the agent notion is the following, "An agent is an encapsulated computational system that is situated in some environment and that is capable of flexible, autonomous action in that environment to meet its design objectives." [3], [4]. Some commonly recognized and important properties of agents include autonomy (they control their behavior), reactivity (they respond and act in a timely fashion), proactiveness (they can take the initiative and opportunistically adopt new goals), locality (agents have a local perception of their environment) and social ability (they can cooperate with other agents to satisfy the designed objectives) [3], [4]. Based on these properties, a multi-agent system can be designed to distribute data and control (decentralization), execute agents asynchronously, be open or closed (agents can freely enter and exit the system), and be composed of homogeneous and/or heterogeneous agents. Not all of these features are necessarily required in all applications and are often

partially present. For example, a multi-agent system may not be homogeneous (a few different types of agents) or may be partially decentralized with a hierarchical structure, etc. In the literature on energy management, a range of strategies using multi-agent systems (MASs) combined with different optimization and machine learning algorithms (to implement decision-making capabilities in the agents) have been suggested.

The concepts of multi-agent systems have been used to tackle a variety of smart grid challenges such as optimal microgrid planning under uncertainty [5], frequency and voltage control [6], power management [7], decentralized control of autonomous polygeneration microgrids [8], coordination of EV charging [9], and optimal energy scheduling for the EVs aggregator [10]. In [11], a hierarchical MAS is implemented to find optimal charging strategies for EVs. No DERs were considered in this study. In [12], a hierarchical MAS is presented to control the EV charging while avoiding congestion in the system. In this study, vehicle-to-grid (V2G) functionality was not considered. Habibdoost et al. [13] have applied hierarchical MAS heuristics to minimize the cost of supporting a micro-grid using EVs, in case of an outage. However, no DSO constraints were considered in this study. In [14], a hierarchical agent-based control system to coordinate the charging of EVs is presented. Mocci et al. [15] have implemented a hierarchical MAS using a quadratic solver to incorporate demand response and coordinated EV charging in distribution grids. However, in all these works, computational complexities remain a problem when applied to large-scale networks. Hence, these solutions, [11-15] may not scale well due to the hierarchical architecture of the designed systems.

Wang et al. [16] have used MAS combined with graph theory to achieve the consensual trade-off between EV charging losses minimization and increased available share of V2G power. Network issues such as congestion management and voltage regulation have not been considered. In [17-19], MASs are modeled to tackle such issues in the presence of storage. However, the computing times of such models increase exponentially as the states in the model increase. In [20], [21], MASs using game theory are designed for efficient management of power flows in buildings with DERs and for optimal allocation of battery storage on the distribution side respectively. However, convergence can be a problem for such systems when applied to a very large-scale network. Shirzeh et al. [22] use a multi-agent system to manage DERs and storage systems for energy management in the smart grid. However, these solutions, [16-22], may not be scalable due to the inherent shortcomings of the applied learning

algorithms for systems with a large number of possible states.

In [23], a scalable MAS is presented for the controlled charging of EVs. However, an accurate model of the distribution system is required by the system. Many of the existing decentralized solutions depend on the availability of an accurate distribution system model to link the charging powers of EVs with voltages and power flows in the network. These necessary models are often inaccurate or unknown, thus hindering or preventing the deployment of these methods in a real-life scenario.

Egbue et al. [24] applied the concept of MAS, by defining simplistic actions for the EV agents, to design a microgrid with a V2G system. No network model is required in the designed MAS, but as a result, the DSO constraints have not been considered. In this work, a self-organized multi-agent system is used to manage a large-scale EV fleet. Self-organized multi-agent systems have been proved to be an efficient approach to deal with complex systems with dynamic requirements in distributed systems. This type of system is also known to be robust, scalable, and reliable. In addition, self-organization enables the engineering of underspecified software (see [25] for more details). According to Serugendo et al. [26], self-organization is “the mechanism or the process enabling a system to change its organization without explicit external command during its execution time”. They considered two types of self-organized systems:

- Strong self-organizing systems are systems where there is no explicit central control either internal or external.
- Weak self-organizing systems are systems where, from an internal point of view, re-organization may be performed under internal (central) control or planning.

The concept of self-organization must be taken in a broad sense, it can concern the neighborhood of agents, the distribution of tasks, goals, etc. Our proposed multi-agent system falls under the category of strong self-organizing systems. Self-organization mechanisms in multi-agent systems can be grouped into three classes: the bio-inspired approaches (such as stigmergy, cooperation, reinforcement, etc.), social-based approaches (trust-based, social functions, auction, etc.), and artificial approaches (authentication chains, tag-based models, and so on) [25]. In this work, we chose to use the theory of adaptive multi-agent system (AMAS) [27] which is based on cooperation as a self-organization mechanism. This approach has been applied successfully to solve problems from several disciplines involving large-scale systems [28]-[31]. Moreover, it is generic and can be combined with other mechanisms of self-organization (such as stigmergy or

reinforcement learning, etc.). The AMAS theory proposes meta-rules and notions allowing self-organization (or self-adaptation) through cooperation. With this theory, we can design a system capable of self-adapting to unseen or unspecified changes in the environment. In this paper, we intend to evaluate the outcomes of a pure AMAS applied to the management of EVs in a large-scale distribution network, before eventually combining it with other mechanisms/techniques.

The proposed adaptive multi-agent system (AMAS) is based on a rule-based feedback control scheme similar to the one that has also been used by Zishan et al. [32] in their MAS. However, to incorporate adaptability in a simple rule-based system, they have utilized reinforcement learning while we are using a different methodology here, i.e., the concepts of cooperation [33]. The behaviors of the agents are defined simply and the system’s objectives are achieved through cooperative interactions among these agents (also the agents and the environment). This simple behavioral design of each agent results in an efficient performance even on a very fast timescale (1 second in this study) and also increases the scalability of the overall system. Additionally, as the system performs online optimization (mainly reactive), it does not need to go through any training phase and thus no statistical approximations or discretization of the action space of each agent is necessary to tackle the curse of dimensionality or to increase the scalability of the designed complex system for real-time operations. Furthermore, in [30], grid challenges involving two actors i.e., prosumers and DSO are considered. However, in this paper, the constraints and the objectives of prosumers, DSO, and BRP are considered altogether. This results in a highly coupled, multiscale problem.

Additionally, not all studies quantify the gap in the optimality of the obtained results to check the quality of the designed methodology. In contrast, to evaluate the performance of the proposed MAS solution, our problem is modeled and solved using mixed integer linear programming (MILP) optimization. Following this, the comparison is drawn in terms of the quality of the solution as well as in terms of the practical computational complexities of both algorithms. The designed system is completely decentralized, scalable, and model-free. We use the term "model-free" in the meaning proposed by Barto and Sutton in [34]. It is considered that the system does not have a model of the environment that allows inferences to be made about how the environment will behave. Instead, the proposed MAS uses trial-and-error interactions with the environment. These characteristics (decentralization, scalability, and model-free) of AMAS

make it a suitable candidate to optimize the operation of a large-scale distribution system while considering a wide range of market actors' constraints. The contributions of this paper can be summarized as follows:

- A model-free AMAS is developed for large-scale distribution networks. The designed system solves the commitment mismatch problem of the BRP by using the connected EVs' batteries, in the presence of photovoltaic (PV) panels, together with satisfying charging constraints of the EVs while avoiding congestion and voltage limits violations in the network. The designed system is completely decentralized, highly scalable, and operates in real-time (1 second in this study).

- A comparison between the proposed AMAS solution and an optimal MILP solution is presented to quantify the performance of the system. The comparison is made both in terms of the distance to optimality of the obtained solution and in terms of the practical computational complexities (time and memory). This comparison paints a more complete picture to evaluate the performance of AMAS for smart grids.

The rest of this paper is organized as follows. In Section 2, the modeling of the problem using MILP and AMAS is presented. Moving forward, the case study to test both systems is presented in Section 3. Finally, the results obtained are discussed in Section 4 and the conclusion is made in the last section.

Modelling

This section presents the detailed mathematical formulation of both MILP and AMAS strategies. The optimization problem formulation is introduced at first along with the relaxations to make the formulation suitable for MILP optimization [35]. Then, the concepts of AMAS and the design of the system are presented.

1.1. Optimization Problem Formulation

1.1.1. Network Physical Constraints

These constraints guarantee the correct power flow results, which are based on the network's physical constraints. The active and reactive powers at each bus are defined as:

$$P_0(t) = P_{grid}(t) \quad (1)$$

$$P_a(t) = P_{PV,a}(t) - P_{load,a}(t) - \tilde{P}_{EV,a}(t) + \Delta P_{EV,a}(t) \quad (2)$$

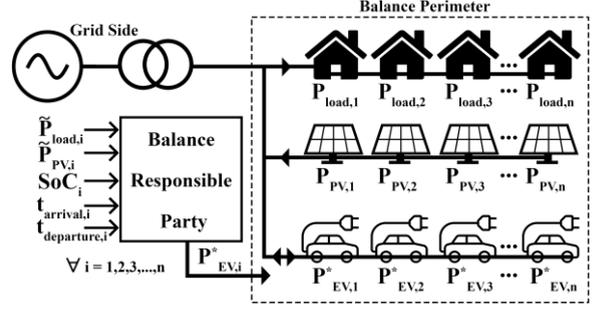


Fig. 1. Diagram of the optimization problem.

where $P_a(t)$ is the active power of Bus a , at instant t . Reactive power flows are modeled similarly. The first bus (Bus 0) of the system is modeled as the grid bus with active power $P_{grid}(t)$. The total power at Bus a , at instant t , is equal to the sum of the injected PV power $P_{PV,a}(t)$, the drawn load power $P_{load,a}(t)$, and the EV power. Term $\tilde{P}_{EV,a}(t)$ is the expected day-ahead active power for the EV connected to Bus a , at instant t , while $\Delta P_{EV,a}(t)$, a decision variable, is the real-time change, with respect to the predicted the day-ahead power, to achieve an optimized solution. The power flowing from Bus a to b is linked to the instantaneous bus voltages as:

$$P_{ab}(t) + jQ_{ab}(t) = V_a(t)(V_a^*(t) - V_b^*(t))Y_{ab}^* \quad (3)$$

where $V_a(t)$ and $V_b(t)$ are the voltages at Bus a and b respectively, and Y_{ab}^* denotes the electrical line's admittance.

1.1.2. DSO Constraints

Active power, reactive power, and bus voltage at each bus should remain within specified limits. Furthermore, there is a limit on the maximum current flowing through an electrical line, $I_{ab,max}$. These limits are stated in (4)-(7).

$$P_{a,min} \leq P_a(t) \leq P_{a,max} \quad (4)$$

$$Q_{a,min} \leq Q_a(t) \leq Q_{a,max} \quad (5)$$

$$V_{a,min} \leq |V_a(t)| \leq V_{a,max} \quad (6)$$

$$|I_{ab}(t)| \leq I_{ab,max} \quad (7)$$

2.1.3. EV Constraints

EV constraints link the variables specific to an EV, such as its state of charge (SoC) and state of health (SoH) with the network flow constraints and introduce limits on these variables. The SoC is calculated as:

$$SoC_a(t) = SoC_a(t-1) + \frac{P_{EV,chg,a}(t)\Delta t}{E_{bat,a}} \eta_{chg} - \frac{P_{EV,dischg,a}(t)\Delta t}{E_{bat,a}} \eta_{dischg} \quad (8)$$

$$SoC_{a,min} \leq SoC_a(t) \leq SoC_{a,max} \quad (9)$$

$$SoC_{a,min,depart} \leq SoC_a(t), \text{ if } t = t_{a,depart} \quad (10)$$

where $P_{EV,chg,a}(t)$ and $P_{EV,dischg,a}(t)$ are the EV charging and discharging powers at instant t respectively. The SoC should always remain within limits specified by $SoC_{a,min}$ and $SoC_{a,max}$. Additionally, (10) specifies that at the time of departure of the EV, the SoC should be higher than a specific value defined by the owner, $SoC_{a,min,depart}$. Terms $E_{bat,a}$, η_{chg} , and η_{dischg} are the energy capacity of EV at Bus a , charging efficiency, and discharging efficiency of the battery respectively. The state of health (SoH) of the battery should be positive and is defined as:

$$SoH_a(t) = SoH_a(t-1) - \frac{P_{EV,a}(t)\Delta t}{0.2E_{bat,a}} \geq 0 \quad (11)$$

The instantaneous SoH depends on the SoH value of the previous instant, and the EV charging or discharging power at instant t . The SoH of a battery helps in assessing its degradation due to battery aging. Its value ranges between 0 and 1. We have considered a capacity fade of 20% as the end-of-life of a battery in this formulation. Once the end-of-life of an EV battery is reached, it will need to be replaced. The total EV charging/discharging power connected to Bus a , at instant t is defined as:

$$P_{EV,a}(t) = P_{EV,chg,a}(t) - P_{EV,dischg,a}(t) = \tilde{P}_{EV,a}(t) + \Delta P_{EV,a}(t) \quad (12)$$

$$0 \leq P_{EV,chg,a}(t) \leq (\rho(t)) P_{EV,a,max} \quad (13)$$

$$0 \leq P_{EV,dischg,a}(t) \leq (1 - \rho(t)) P_{EV,a,max} \quad (14)$$

where $\rho(t)$ is a binary variable to ensure that an EV does not charge and discharge simultaneously at instant t , and $P_{EV,a,max}$ is the maximum charging/discharging power of the EV.

2.1.4. Objective Function

The objective is to minimize the difference between the real-time energy consumption/production in the BRP perimeter and the commitment value for the studied imbalance settlement period. The objective function is defined as:

$$obj : \min_{\Delta P_{EV,a}(t)} \sum_{N=1}^{N_{end}} \left| \sum_{t=1}^{\Delta T} \left((\tilde{P}(N) - P_{BRP}(t, \Delta P_{EV})) \Delta t \right) \right| \quad (15)$$

where N represents the present imbalance settlement period, ΔT is the length of the imbalance settlement period, and $t \in [1, \Delta T]$ is the smaller time resolution within this imbalance settlement period. Term $\tilde{P}(N)$ is the day-ahead planned power for the imbalance settlement period N , and $P_{BRP}(t, \Delta P_{EV})$ is the total real-time consumption/production in the BP. The planned BRP commitment values $\tilde{P}(N)$ are calculated as:

$$\tilde{P}(N) = \frac{1}{\Delta T} \sum_{t=1}^{\Delta T} \left(mean_{1hr}(\tilde{P}_{PV}(t)(1 + e_{PV}(t))) - mean_{10min}(\tilde{P}_{load}(t)(1 + e_{load}(t)) - mean_{10min}(\tilde{P}_{EV}(t))) \right) \quad (16)$$

where $\tilde{P}_{PV}(t)$, $\tilde{P}_{load}(t)$, and $\tilde{P}_{EV}(t)$ are the day-ahead forecasted PV, load, and EV profiles respectively. Errors in the PV forecast $e_{PV}(t)$, and the load forecast $e_{load}(t)$ are also modeled here. In (16), two different mean values can be observed, $mean_{1hr}$ and $mean_{10min}$, indicating the means of a time series over one hour and ten minutes respectively. This is because, usually the forecasts for PV are provided as one-hour average values while in the case of loads, smart meters can provide measurements at a resolution of ten minutes.

2.1.5. MILP Approximations

The original formulation is nonlinear, due to the product of voltages in (3). Thus, relaxations are made to transform it into a problem compatible with MILP. To linearize, the small angle approximation (i.e., $\sin(\theta_a - \theta_b) \approx (\theta_a - \theta_b)$), is considered here. Additionally, it is also assumed that the per unit values of the voltages are sufficiently close to 1 (i.e., $|V| \approx 1$). This results in two separate equations for active and reactive powers, (17) and (18) respectively.

$$P_{ab}(t) = G_{ab}(V_a(t) - V_b(t)) + B_{ab}(\theta_a(t) - \theta_b(t)) \quad (17)$$

$$Q_{ab}(t) = B_{ab}(V_a(t) - V_b(t)) + G_{ab}(\theta_b(t) - \theta_a(t)) \quad (18)$$

where conductance and susceptance values of the electrical line between Buses a and b are denoted by G_{ab} and B_{ab} respectively. Terms θ_a and θ_b are the phase angles at Bus a and Bus b respectively. Combined with the objective function and the constraints described earlier, this formulation can be directly solved using MILP.

2.2. Adaptive Multi-Agent System (AMAS)

A MAS is a computerized system, in which a group of intelligent agents are continuously observing an environment and solving a set of complex problems through interactions. These reactive agents are autonomous and help to achieve decentralization of the system. An agent executes in a loop and goes through the following three stages in its life cycle:

1. Perception stage: the agent gathers data from its environment.
2. Decision stage: based on the observed data, an intelligent decision is made.
3. Action stage: the previously selected decision is implemented by taking the required actions.

An AMAS is a specific type of MAS, in which decision-making agents are communicating with the neighboring agent(s) to achieve a common goal through cooperative interactions, instead of dividing a big problem into smaller sub-problems. Each agent has its objective and a set of actions. At every instant, each agent decides whether to pursue its own goal or to help neighboring agent(s) through cooperative actions. To differentiate the notion of a bad action from a good action, each agent holds a criticality value defined between 0 and 1. A near zero criticality value would indicate the success of an agent in achieving its objective, and vice versa. The choice of action by each agent is based on the comparison between these criticalities i.e., an agent may help a neighboring agent in case the latter is more critical than the former.

The “agentification processes” is defined as the mapping of a physical distribution network’s elements (such as electrical buses, lines, EVs, etc.) to software agents of the multi-agent system (such as bus agents, line agents, EV agents, etc.). In the proposed design, each physical component of the smart grid is a cooperative agent in the software domain. The designed system consists of four types of intelligent agents: bus agents, line agents, BRP agents, and EV agents. The design of each agent type is described in the following section. The “agentified model” of the sub-section of a distribution network is shown in Fig. 2. Each agent communicates the criticality value only with its neighboring agents. The BRP agent only communicates its criticality with all EV agents present in the BRP perimeter. The BRP agent communicates with the TSO to submit its day-ahead power consumption/production schedule.

As a comparative analysis will be performed between different strategies, it is important to mention that the

considered MILP is deterministic and centralized (i.e., SoC, arrival time, departure time, etc. are considered as known by the centralized agent). On the contrary, the agents in an AMAS react to the variations in the environment (e.g., energy commitment mismatches for the BRP) and the requests of other neighboring agents in real-time. Hence, only a limited knowledge of the overall problem is required by them. These two approaches will also be compared with a so-called “dumb” (uncontrolled) charging strategy, where EVs charge at a constant power between their arrival and their departure times.

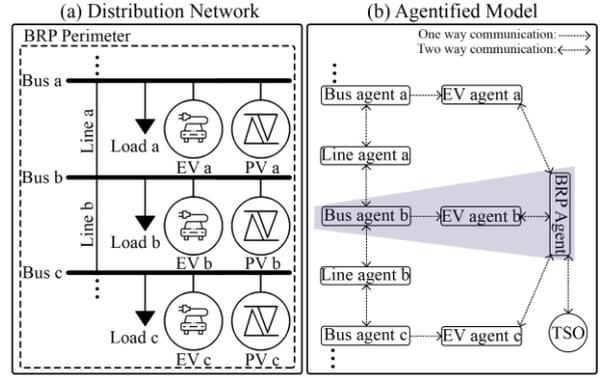


Fig. 2. a) Sub-section of a distribution network b) “Agentified model” (the shaded area indicates the neighborhood of the EV agent b) of the distribution network in (a).

2.2.1. Line Agent

Line agents are designed to prevent line congestion. The neighborhood of a line agent consists of the bus agents corresponding to the electrical buses connected to it. Each line agent calculates its criticality based on the magnitude and direction of the current through the electrical line. In case of a non-zero criticality, the agent will request the EVs for cooperation. Line criticality is calculated as:

$$Cr_l(t) = \begin{cases} 0, & \text{if } I(t) \leq I_{th} \\ \frac{I(t) - I_{th}}{I_{max} - I_{th}}, & \text{if } I(t) > I_{th} \end{cases} \quad (19)$$

where at instant t , $Cr_l(t)$ is the agent’s criticality, and $I(t)$ is the current through the electrical line. Term I_{max} is the maximum current allowed through the line, which is set arbitrarily to 95% of the rated line current value. This 5% margin helps to avoid congestion arising due to sudden current peaks. I_{th} is the current threshold (arbitrarily set to $0.7I_{max}$), above which the criticality starts increasing linearly. As in (19) the line agent’s criticality is dependent

only on the instantaneous value of the electrical current, therefore an extreme behavior by the agent (continuous switching between highly critical and non-critical states) can be expected in case of congestion. These oscillations are undesirable especially when the agents are reacting every second. To tackle this, a criticality based on a memory term is introduced and named the ‘‘incremental criticality’’. Once a line becomes critical, the agent uses the incremental criticality value, $Cr_{i,l}(t)$, to request cooperative actions. This incremental criticality is defined as:

$$Cr_{i,l}(t) = K_l Cr_l(t) + (1 - K_l) Cr_{i,l}(t-1) \quad (20)$$

The incremental criticality converges towards the effective criticality value $Cr_l(t)$. The rate of convergence is controlled by the parameter K_l . Larger values of K_l result in quicker convergence but increase the oscillations in the system. Smaller values ensure none or minimal oscillations but at the cost of slower convergence. As this incremental criticality is communicated to the neighboring agents once line congestion is detected, the electrical line current does not show extreme changes in its magnitude from one instant to the next.

2.2.2. Bus Agent

Bus agents are designed to prevent voltage limit violations. The neighborhood of a bus agent consists of the lines and EVs agents corresponding to the electrical lines and EVs connected physically to that bus. At first, each bus agent tries to prevent its voltage limit violation locally through cooperation with the EV(s) connected to that bus. When it is not possible, the bus agent starts communicating with other EVs in the distribution network to maintain its voltage level. Bus agent’s criticality is calculated as:

$$Cr_b(t) = \begin{cases} 0, & \text{if } |(V(t) - V_u)| \leq V_{th} \\ \frac{|V(t) - V_u| - V_{th}}{|V_{max} - V_u| - V_{th}}, & \text{if } |(V(t) - V_u)| \geq V_{th} \end{cases} \quad (21)$$

where $Cr_b(t)$ is the bus agent’s criticality, V_u is the nominal voltage, V_{max} corresponds to the maximum percentage deviation allowed in the bus voltage, and V_{th} is the threshold for the percentage deviation in the bus voltage, above which the bus criticality starts increasing linearly. To prevent the oscillations, similar to (20), incremental bus criticality (with K_b as the control parameter) is introduced as:

$$Cr_{i,b}(t) = K_b Cr_b(t) + (1 - K_b) Cr_{i,b}(t-1). \quad (22)$$

2.2.3. BRP Agent

The BRP agent’s goal is to minimize the commitment mismatch for the current imbalance settlement period N , similar to the modeled objective function for the MILP optimization formulation. The BRP agent communicates with all the EVs present in its perimeter. As AMAS is an iterative system, the instantaneous criticality of the BRP agent, during an imbalance settlement period N , depends on the cumulative sum of the commitment mismatch observed since the start of the imbalance settlement period N . Its criticality is defined as:

$$Cr_{BRP}(t) = \frac{\left| \sum_{j=t_{start}}^t ((\tilde{P}(N) - P_{BRP}(j)) \Delta t) \right|}{tr_{period} K_{BRP}} \quad (23)$$

where t_{start} is the starting time of the current settlement period, K_{BRP} is the BRP agent’s tuning parameter and tr_{period} is the remaining time before the end of the current imbalance settlement period. The execution steps of each line, bus, and BRP agent, during a single agent cycle, are summarized in Algorithm 1. The execution steps of each EV agent are summarized separately in Algorithm 2, as the decision and the action stages of the EV agents differ from the decision and the action stages of the types of agents presented in Algorithm 1.

2.2.4. EV Agent

Each EV agent manages the consumption of its corresponding EV to attain the desired minimum SoC before its departure, while also cooperating with other neighboring agents for their objectives. The EV decision i.e., the power at which an EV should charge or discharge at instant t , is made through a comparison of criticalities i.e., the EV will help neighboring agents which are more critical than itself, otherwise, it will continue charging. The criticality $Cr_{EV,a}(t)$ of the EV agent a is calculated as:

$$Cr_{EV,a}(t) = \frac{(SoC_{a,min,depart} - SoC_a(t)) E_{bat,a}}{t_{a,depart} P_{EV,a,max}}. \quad (24)$$

Algorithm 1: Each line, bus, and BRP agent's execution steps

```

begin agent cycle
  /* Perception Stage
  Requests ← received criticality messages from
  neighboring agents
  Cr ← calculate the agent's criticality
  /* Decision stage
  if ((Cr > 0) or (Requests ≠ null)) then
    if (Requests ≠ null) then
      if (Cr < criticality of the most critical
      request) then
        Request to forward ← most critical
        request received
      else
        Request to forward ← agent's criticality
        request
      else
        Request to forward ← agent's criticality
        request
    /* Action stage
    if (Request to forward ≠ null) then
      Send "Request to forward" to all neighboring
      agents
  end agent cycle

```

This EV criticality is compared with the criticalities of the neighboring agents. If the EV agent has the highest criticality, it continues charging at the planned charging power, otherwise, it helps the most critical neighboring agent(s). It is crucial to design the EV agent such that it can handle the antagonist requests (when two or more agents request opposite cooperative actions from the same EV agent). For example, a BRP agent can request the EVs to consume more when the commitment mismatch is positive, while simultaneously a line agent can request the EVs to charge less so that the congestion can be avoided. None of these requests can be completely ignored, hence the updated EV power at instant t is calculated as:

$$P_{EV,a}(t) = P_{EV,a}(t-1) \pm (Cr_{max}(t) - hCr_{ant}(t))P_{EV,a,max} \quad (25)$$

where $Cr_{max}(t)$ is the criticality of the most critical request and $Cr_{ant}(t)$ is the greatest criticality corresponding to an antagonist request, at instant t . Term h is a weighting function for the antagonist requests. Function h is defined as:

$$h(Cr_{max}(t)) = \begin{cases} 1 + \frac{\alpha - 1}{T_{l,min}} Cr_{max}(t), & \text{if } Cr_{max}(t) > T_{l,min} \\ e^{\left(\frac{Cr_{max}(t)}{K_a + K_b}\right)}, & \text{if } T_{l,min} \leq Cr_{max}(t) \leq T_{l,max} \\ 1 - \frac{\beta}{1 - T_{l,max}} Cr_{max}(t), & \text{if } Cr_{max}(t) > T_{l,max} \end{cases} \quad (26)$$

where $T_{l,min}$ and $T_{l,max}$ are two threshold values. Between these two thresholds, function h behaves exponentially, and decreases linearly otherwise. Parameters K_a and K_b depend on the values of α , β , $T_{l,min}$ and $T_{l,max}$. It can be observed in Algorithm 2 that the decision and action stages, during an EV agent's cycle, are different from that of all other types of intelligent agents in the system. This is because the instantaneous charging or discharging power of the EV is also calculated, through a comparison of criticalities, during the decision and the action stages.

Algorithm 2: Each EV agent's execution steps

```

begin agent cycle
  /* Perception Stage
  Requests ← received criticality messages from
  neighboring agents
  Cr ← calculate the agent's criticality
  /* Decision stage
  if ((Cr > 0) or (Requests ≠ null)) then
    if (Requests ≠ null) then
      if (Cr < criticality of the most critical
      request) then
        Cr_max ← criticality of the received most
        critical request
        Issue_max ← type of issue corresponding
        to the maximum criticality
      else
        Cr_max ← EV agent's criticality
        Issue_max ← type of issue corresponding
        to the maximum criticality
      else
        Cr_max ← EV agent's criticality
        Issue_max ← type of issue corresponding
        to the maximum criticality
    Cr_ant ← maximum criticality of the antagonist
    request(s)
    h ← calculate using (26)
    if (Issue_max ∈ {BRP under-consumption, bus
    over-voltage, high export line current}) then
      Power_EV ← Power_EV at previous instant +
      (Cr_max - h * Cr_ant) * maximum charging
      power

```

```

if ( $Issue_{max} \in \{BRP \text{ over-consumption, bus}$ 
under-voltage, high import line current}) then
     $Power_{EV} \leftarrow Power_{EV}$  at previous instant -
    ( $Cr_{max} - h * Cr_{ant}$ ) * maximum charging
    power
    /* Action stage
    Set  $Power_{EV}$  as instantaneous EV power
end agent cycle

```

2. Case Study

2.1. Distribution Network

A large-scale distribution network is modeled using the ‘‘IEEE European Low Voltage Test Feeder’’ (LVTF) model [36]. A graphical representation of the network is shown in Fig. 3. The modeled larger-scale distribution network consists of 3 districts, where each district contains 3 sub-districts. In total, the network consists of 9 subdistricts (A, B, C, D, E, F, G, H, and I), each representing a single IEEE LVTF model. The complete distribution network includes 495 household loads, along with 495 PVs and 495 EV connections on the same load buses. Line 1 (highlighted in red in Fig. 3), which connects sub-district A to the grid, is selected for the congestion study.

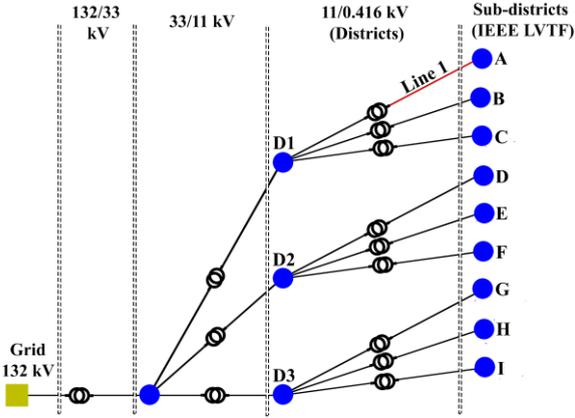


Fig. 3. Line diagram of the studied distribution network

2.2. Implementation

2.2.1. MILP

CVXPY is used to solve the MILP formulation [37]. The computing machine is equipped with an Intel i5-10210 CPU along with 8 GB of DDR4 random-access memory (RAM). The secondary storage is an NVM express

(NVMe) type drive. The temporal resolution of the MILP study is set to 1 minute.

2.2.2. AMAS

The proposed AMAS system is developed in JAVA and is called ADEMIS (ADaptive Energy Management In Smart Grids) [38]. It is interfaced with DiGSILENT PowerFactory (PF) to perform the load flow calculations [39]. The system executes in a loop. Communication between JAVA and PF is performed through a Python script. The time resolution of this system is 1 second. There are 495 EV agents, 1 BRP agent, 1857 bus agents, and 1855 line agents in the system, making a total of 4208 intelligent AMAS agents.

2.3. Datasets

The individual household consumption data is provided with the IEEE LVTF network model. The length of each load profile is 1 day with a 1-minute resolution. The irradiance data, obtained from the database of the National Renewable Energy Laboratory (NREL), is used for PV profiles [40]. The resolution of the PV data is 1 second. The total power generated by a group of PV panels belonging to a given household at time t , $P_{PV}(t)$, is calculated as:

$$P_{PV}(t) = A\eta_{PV}Irr(t) \quad (27)$$

where A is the area of the PV panels, η_{PV} is the efficiency of the PV panels, $Irr(t)$ is the irradiance value at time t . The area of the PV panels and the efficiency of the PV panels for this study are equal to 10 m^2 and 0.17. The duration of this study is one hour (4 imbalance settlement periods of 15 minutes each). The arbitrarily selected hour for this study is from 16:00 till 17:00. The forecast errors follow a Gaussian distribution with a standard deviation of 20% and zero mean value, as a first approach.

2.4. EV Models

All EVs hold the same battery capacity of 30 kWh and a maximum power rating of 7 kW . The minimum and maximum SoCs are set to 0.3 and 0.8 respectively; while each EV should have at least 0.7 SoC at the time of its departure. A large number of EVs are arriving at the start of period 1 and departing near the end of the period (23 EVs in sub-district A of the distribution network). These EVs, due to their short connection time with the grid, will demand a large amount of power, thus causing congestion

in Line 1. Other grid-connected EVs, depending on their constraints, will try to minimize the arising problems in the grid.

2.5. Antagonist Issues

The selected PV profile results in an excess of production during the first two imbalance settlement periods, while a lack of production exists in the two remaining periods, shown in Fig. 4. As the real-time production is different from the day ahead forecast due to forecast errors, the BRP commitment mismatch is non-zero in all periods. Also, as mentioned earlier, line congestion will occur in Line 1 during the first imbalance settlement period if the dumb strategy is applied. The value of the rated current is obtained based on the electrical line's type (provided in the IEEE LVTF network model) and is equal to 0.21 kA. As both the line congestion and an excess PV production in the sub-district A is occurring during the first period, the EVs present in the sub-district A will therefore face an antagonistic scenario: they will be encouraged to consume more by the BRP to minimize the commitment mismatch while the line will encourage them to consume less to avoid congestion. These types of situations may arise in a real-life distribution network, thus consideration of such issues in this case study makes it incline more toward a realistic scenario.

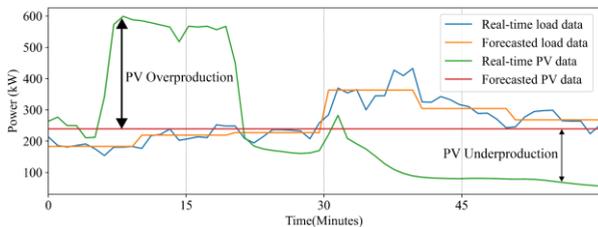


Fig. 4. Load and PV data for the case study.

3. Results

In this section, the quality of the solutions obtained is compared and also the computational complexities of the MILP and AMAS solutions are discussed.

3.1. Solutions Quality Comparison

The electrical current profile of Line 1 is shown in Fig. 5. A significant level of congestion can be seen for the uncontrolled “dumb” charging strategy. The congestion lasts for 30.16% of the imbalance settlement period with an average value equal to 110.5% of the rated line current.

Line congestion is avoided in both MILP and AMAS solutions. The total power consumption of all the EVs is also shown in Fig. 5. During the first two imbalance settlement periods, an excess of the PV production with respect to the forecasted values leads to the EVs charging more compared to their planned consumption to achieve the objective of the BRP (for both MILP and AMAS). For the third and fourth imbalance settlement periods, an under-production by the PVs with respect to the forecasted values results in reduced total consumption by the EVs for the MILP and the AMAS strategies.

Table 1 shows the comparison of the BRP commitment mismatch values for the uncontrolled charging, the AMAS, and the MILP strategies. As expected, the uncontrolled charging of the EVs leads to the highest mismatch, with a total of 134.32 kWh. MILP is providing the optimal solution with the total commitment mismatch equal to zero. AMAS is performing exceptionally well compared to the uncontrolled charging strategy, reducing its total commitment mismatch by 99.5%. Both MILP and AMAS solutions are satisfying all the earlier mentioned DSO and prosumers constraints.

Table 1. Commitment Mismatch Comparison (kWh).

Period	Uncontrolled Charging	AMAS	MILP
Period 1	38.061	0.015	0
Period 2	11.125	0.012	0
Period 3	37.202	0.001	0
Period 4	47.932	0.630	0
Total	134.32	0.658	0

Unlike MILP, AMAS is reactive and not predictive. But it still manages to follow the path obtained by the MILP solution for the total consumption of all EVs. This ensures that not only AMAS can find near-optimal solutions but also, it can be scaled to obtain efficient solutions for larger networks as the complexities of centralized predictive strategies increase dramatically with the number of system variables. As the AMAS is purely reactive, without any anticipative capabilities, a relatively oscillatory solution profile can be observed. Multi-agent learning is envisaged to further improve the performance of the designed AMAS platform.

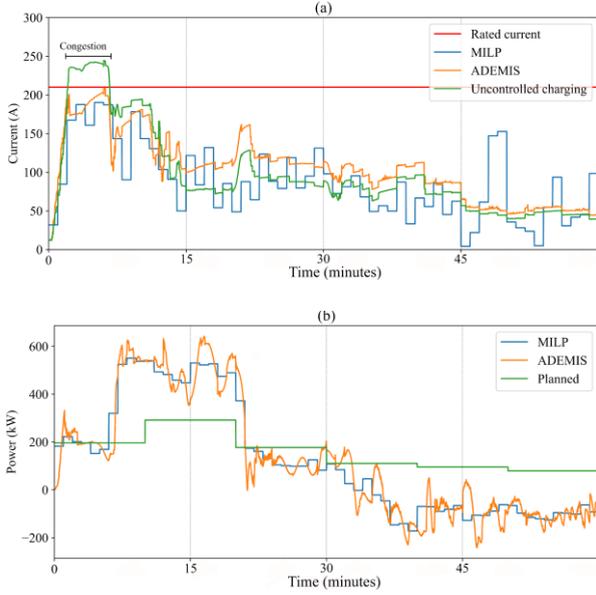


Fig. 5. Case study results: a) Line 1 current b) Total EVs power.

3.2. Computational Complexity Comparison

The computational complexities of the MILP strategy, obtained by averaging 10 independent runs for each data point, are shown in Fig. 6. Curve fitting reveals that the practical time complexity big O of the algorithm equals n^2 , where n is the problem size. For the centralized MILP, there are no differences between the simulation or the real-life complexities, as a single node would be processing all the information in both scenarios. But for the decentralized AMAS, as the computing device used for simulation studies generally has a single processing unit and in a real-life implementation, each agent is supposed to have its processing unit, thus performing simulations on a single computing device can be more restrictive. Curves for both the simulation complexities and the real-life implementation complexities for AMAS are given in Fig. 6.

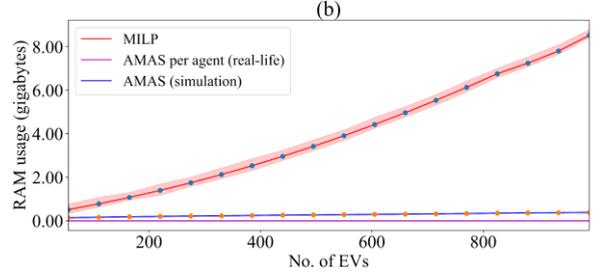
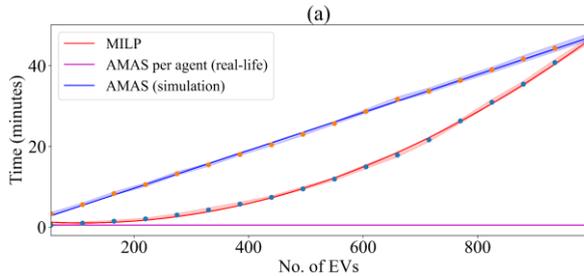


Fig. 6. Computational complexities comparison a) Time complexities b) Memory complexities.

In the context of practical implementation, as the computing time per agent in an AMAS system is constant regardless of the number of total agents in the system, our proposed decentralized system outperforms the MILP approach even at a small scale. This is a significant result of this paper: our proposed method allows us to find a near-optimal solution, given by the MILP approach here, with a fraction of its computing time. The processing times per agent of all intelligent agents in the AMAS are presented in Table 2. In the context of the simulation studies, a linear relationship is observed for the AMAS and hence it will outperform the MILP for larger networks.

Table 2. AMAS per agent processing times

Agent	Processing Time (ms/iteration)
Bus	0.003
Line	0.004
BRP	0.064
EV	0.540

A significant difference can also be observed in terms of memory consumption as the centralized MILP, being predictive, processes larger amounts of input data at once compared to the reactive AMAS, in which each agent processes only a limited amount of the local data at each time step. This would pose a problem if the required memory by the MILP approach to perform the optimization is greater than the available RAM of the computing device. On the specified computing device, this happens when the MILP optimization with 1000 or more EVs is performed. Modern operating systems (OSs) tackle this problem by creating virtual storage in the secondary memory using memory paging [41], but this comes at the cost of an increment in the running time as the secondary storages are much slower compared to the RAM. To get an

idea of the impact of the available memory on the time complexity, a comparison of the MILP optimization times (with and without the memory constraint) is presented in Fig. 7. The MILP algorithm time with no memory constraint is observed simply by assuming there is enough RAM available to process the entirety of the input data. However, if the memory constraint is considered for more complex and large-scale problems, the operating system has to start using the slower secondary memory storage when the size of the data to process to perform optimization is greater than the available RAM. The algorithm time with the memory constraint is calculated as:

$$t_{cstr} = \left(\min(r_m, 1) + \left((1 - \min(r_m, 1)) r_s \right) \right) t_{noncstr} \quad (28)$$

where $t_{noncstr}$ is the algorithm time when no memory constraint is considered, r_m is the ratio of the available RAM over the total memory required by the MILP strategy, and r_s is the memory speed ratio, which is obtained by dividing the input-output operations per second (IOPS) of the RAM over the IOPS of the installed secondary storage. Generally, OSs store working data as small chunks at random locations in the storage memory. Thus, benchmarks through random read/write cycles with a file size of 4 *KiB* are performed to obtain the average IOPS values. These benchmarks show that the IOPS of the RAM and that of the installed secondary storage are equal to 0.259 million and 0.014 million respectively, on the specified computing machine.

The memory-adjusted time complexities (logarithmic vertical scale) are shown in Fig. 7. The red region corresponds to the extrapolated time complexities, while the green region corresponds to the measured computing times. For the decentralized AMAS, two curves are presented in Fig. 7. One curve includes only the total time taken by all AMAS agents (blue curve), while the other curve considers the time spent in executing the load-flows as well (green curve). Memory constraint is not a problem for AMAS studies here due to low memory consumption. Extrapolation of the memory consumption suggests that the designed system can perform simulation studies with up to 25,000 EVs approximately before the bottleneck of the RAM installed on the specified computing machine is reached.

It should be noted that the MILP time complexity is highly dependent on the secondary storage technology for larger distribution networks. This considerable shift of the solution time without memory constraint (red curve) to the time with memory constraint (black curve), in Fig. 7,

favors the use of the decentralized AMAS for simulating large-scale networks above 1000 EV agents, compared to the centralized MILP. It may be argued that a decentralized version of the MILP problem (not analyzed here) would lead to lower computing times. However, decentralized MILP is generally not used for such a high number of agents as the computing time remains quite prohibitive. Furthermore, the application of such methodologies relies on the availability of an accurate distribution system model, which is not always present.

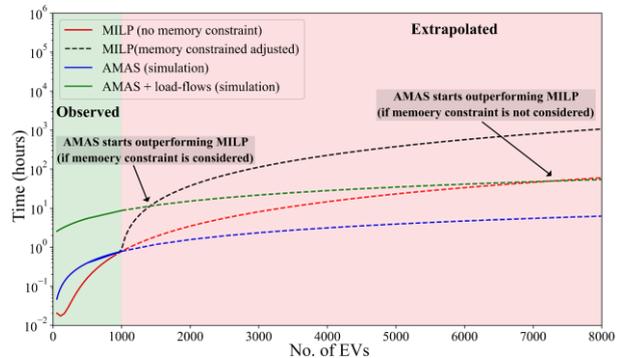


Fig. 7. Computing times comparison against the total number of EVs in the system.

4. Conclusion

This paper proposes an adaptive multi-agent platform for solving a three-level management problem involving a BRP, a DSO, and EV owners. It also compares the results of the centralized MILP with the presented AMAS approach. In the context of a real-life implementation, it can be observed that the AMAS results are near-optimal compared to the optimal MILP approach, for only a fraction of its computing time. The proposed method is also observed to be far more efficient than a “dumb” uncontrolled strategy as it reduces the energy commitment mismatch by more than 99%. In the context of simulating large-scale networks, it can be assumed that the AMAS would outperform the MILP (with memory constraints), in terms of computing times, for systems including more than 1000 EVs. In terms of memory usage, the AMAS can simulate up to 25 times more agents than the MILP approach without exceeding the primary storage capacity of the computer used in this study. These AMAS characteristics make it an ideal candidate both as a practical, real-life solution and to carry out optimization studies for large-scale networks. In the future, a more realistic comparative analysis will be carried out between the AMAS and a stochastic MILP problem, as both would be intended to handle uncertainty. This is expected to lead

to considerably greater computing times for the MILP problem compared to its deterministic version, as in stochastic optimization, several scenarios must be considered. This comparison should make the advantages of the AMAS even more clear.

References

- [1] A. Dorri, S. S. Kanhere and R. Jurdak, "Multi-Agent Systems: A Survey," in *IEEE Access*, vol. 6, pp. 28573-28593, 2018, doi: 10.1109/ACCESS.2018.2831228.
- [2] S. J. Russell (Stuart Jonathan), *Artificial Intelligence: A Modern Approach*. Upper Saddle River, N.J.: Prentice Hall, 2010.
- [3] D. Ye, M. Zhang and A. V. Vasilakos, "A Survey of Self-Organization Mechanisms in Multiagent Systems," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 3, pp. 441-461, March 2017, doi: 10.1109/TSMC.2015.2504350.
- [4] M. Woolridge, *Introduction to Multiagent Systems*. John Wiley & Sons, Inc., USA, 2001.
- [5] S. Mishra, C. Bordin, A. Tomasgard, I. Palu, "A multi-agent system approach for optimal microgrid expansion planning under uncertainty", *International Journal of Electrical Power & Energy Systems*, vol. 109, pp. 696-709, 2019.
- [6] H. K. Vanashi, F. D. Mohammadi, V. Verma, J. Solanki, S. K. Solanki, "Hierarchical multi-agent based frequency and voltage control for a microgrid power system", *International Journal of Electrical Power & Energy Systems*, vol. 135, 2022.
- [7] A. M. Abdulmohsen, W. A. Omran, "Active/reactive power management in islanded microgrids via multi-agent systems", *International Journal of Electrical Power & Energy Systems*, vol. 135, 2022.
- [8] C.S. Karavas, G. Kyriakarakos, K. G. Arvanitis, G. Papadakis, "A multi-agent decentralized energy management system based on distribute intelligence intelligence for the design and control of autonomous polygeneration microgrids", *Energy Conversion and Management*, Volume 103, pp. 166-179, 2015.
- [9] V. Boglou, C. S. Karavas, K. Arvanitis, A. Karlis, "A Fuzzy Energy Management Strategy for the Coordination of Electric Vehicle Charging in Low Voltage Distribution Grids.", *Energies*, vol. 13, 2020.
- [10] M. W. Khan, J. Wang, "Multi-agents based optimal energy scheduling technique for electric vehicles aggregator in microgrids", *International Journal of Electrical Power & Energy Systems*, vol. 134, 2022.
- [11] E. L. Karfopoulos, and N. D. Hatzigiorgiou, "A Multi-Agent System for Controlled Charging of a Large Population of Electric Vehicles," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1196-1204, 2013.
- [12] J. Hu, H. Morais, M. Lind, and H. W. Bindner, "Multi-agent based modeling for electric vehicle integration in a distribution network operation," *Electric Power Systems Research*, vol. 136, pp. 341-351, 2016.
- [13] M. Habibidoost, and S. M. T. Bathaee, "A self-supporting approach to EV agent participation in smart grid," *International Journal of Electrical Power Energy Systems*, vol. 99, pp. 394-403, 2018.
- [14] P. Papadopoulos, N. Jenkins, L. M. Cipcigan, I. Grau, and E. Zabala, "Coordination of the Charging of Electric Vehicles Using a Multi-Agent System," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 1802-1809, 2013.
- [15] S. Mocci, N. Natale, F. Pilo, and S. Ruggeri, "Multi-agent control system to coordinate optimal electric vehicles charging and demand response actions in active distribution networks," in *RPG*, Naples, 2014, pp. 1-6.
- [16] L. Wang, and B. Chen, "Distributed control for large-scale plug-in electric vehicle charging with a consensus algorithm," *International Journal of Electrical Power Energy Systems*, vol. 109, pp. 369-383, 2019.
- [17] F. Lauri, G. Basso, J. Zhu, R. Roche, V. Hilaire, and A. Koukam, "Managing Power Flows in Microgrids Using Multi-Agent Reinforcement Learning," *Agent Technologies for Energy Systems*, 2013.
- [18] L. Raju, R. S. Milton, and A. A. Morais. "Autonomous Energy Management of a Micro-Grid using Multi Agent System.", *Indian Journal of Science and Technology*, vol. 9, no. 13, 2016.
- [19] E. Shirazi, and S. Jadid, "A multiagent design for self-healing in electric power distribution systems," *Electric Power Systems Research*, vol. 171, pp. 230-239, 2019.
- [20] H. Huang, Y. Cai, H. Xu and, H. Yu, "A Multiagent Minority-GameBased Demand-Response Management of Smart Buildings Toward Peak Load Reduction," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 36, no. 4, pp. 573-585, 2017.
- [21] Y. Zheng, D. J. Hill, and Z. Y. Dong, "Multi-Agent Optimal Allocation of Energy Storage Systems in Distribution Systems," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 4, pp. 1715-1725, 2017.
- [22] H. Shirzeh, F. Naghdy, P. Ciufo, and M. Ros, "Balancing Energy in the Smart Grid Using Distributed Value Function (DVF)," *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 808-818, 2015.
- [23] E. Ucer, M. C. Kisacikoglu, M. Yuksel, and A. C. Gurbuz, "An InternetInspired Proportional Fair EV Charging Control Method," *IEEE Systems Journal*, vol. 13, no. 4, pp. 4292-4302, Dec. 2019.
- [24] O. Egbue, and C. Uko, "Multi-agent approach to modeling and simulation of microgrid operation with vehicle-to-grid system," *The Electricity Journal*, vol. 33, no. 3, pp. 10674, 2020.
- [25] G. Di Marzo-Serugendo, M.P. Gleizes, and A. Karageorgos, "Self-Organisation and Emergence in MAS: An Overview.," *Informatica*, 30(1):45--54, 2005
- [26] G. Di Marzo-Serugendo, M.P. Gleizes, and A. Karageorgos, *Self-organising Software*, Springer Berlin Heidelberg, 2011, Natural Computing Series book series (NCS), 978-3-642-17347-9.
- [27] S. Zafar, V. Maurya, A. Blavette, H. B. Ahmed, G. Camilleri, and M. Gleizes, "Adaptive Multi-Agent System and Mixed Integer Linear Programming Optimization Comparison for Grid Stability and Commitment Mismatch in Smart Grids," in *ISGT*, Espoo, Finland, 2021.
- [28] N. Couellan, T. Jorquera, J.P. Georgé, and S. Jan, "Self-Adaptive Support Vector Machine: A Multi-Agent Optimization Perspective", *Expert Systems with Applications*, Elsevier, 2015, 42 (9), pp.4284-4298.
- [29] A. Perles, G. Camilleri, M.P. Gleizes, O. Chilard, and D. Croteau, "Principle and Evaluation of a Self-Adaptive Multi-Agent System for State Estimation of Electrical Distribution Network", *World Congress on Sustainable Technologies (WCST 2016)*, Dec 2016, London, United Kingdom. pp 17-22
- [30] D. A. Guastella, G. Marcillaud, and C. Valenti, "Edge-Based Missing Data Imputation in Large-Scale Environments" *Information* 12, no. 5: 195, 2021. <https://doi.org/10.3390/info12050195>
- [31] G. Marcillaud, V. Camps, S. Combettes, M. P. Gleizes, and E. Kaddoum, "A Self-Adaptive Module for Cross-Understanding in

- Heterogeneous MultiAgent Systems”, 13th International Conference on Agents and Artificial Intelligence (ICAART 2021), Feb 2021, Portugal. pp.353-360.
- [32] A. A. Zishan, M. M. Haji and, O. Ardakanian, “Adaptive Congestion Control for Electric Vehicle Charging in the Smart Grid,” IEEE Transactions on Smart Grid, vol. 12, no. 3, pp. 2439-2449, 2021.
- [33] M.P. Gleizes, V. Camps, and P. Glize, “A theory of emergent computation based on cooperative self-organization for adaptive artificial systems,” Fourth European Congress of Systems Science, Valencia, Spain, 1999.
- [34] R. S. Sutton, and A. G. Barto, Reinforcement Learning: An Introduction. Second: The MIT Press, 2018
- [35] J. A. Taylor, Convex Optimization of Power Systems. Cambridge: Cambridge University Press, 2015, pp. 44-50.
- [36] IEEE Distribution Test Feeders, Accessed on Oct. 11, 2021. [Online]. Available: <http://sites.ieee.org/pestestfeeders/>
- [37] S. Diamond, and S. Boyd, “CVXPY: A Python-embedded modeling language for convex optimization,” Journal of Machine Learning Research, vol. 17, no. 83, pp. 1-5, 2016.
- [38] J. Blanc-Rouchosse, A. Blavette, H. B. Ahmed, G. Camilleri, and M. P. Gleizes, “Multi-Agent System for Smart-Grid Control with Commitment Mismatch and Congestion,” in ISGT, Bucharest, Romania, 2019, pp. 1-5.
- [39] DIgSILENT PowerFactory. Accessed on Oct. 11, 2021. [Online]. Available: <https://www.digsilent.de/en/powerfactory.html>
- [40] M. Sengupta, and A. Andreas, (2010). “Oahu Solar Measurement Grid (1-Year Archive): 1-Second Solar Irradiance,” Oahu, Hawaii (Data), NREL Report No. DA-5500-56506, 2010. [Online]. Available: <http://dx.doi.org/10.5439/1052451>
- [41] H. Deitel, An Introduction to Operating Systems. Addison-Wesley Publishing Company, 1983, pp. 424-441.