Demand Response as a Service: Clearing Multiple Distribution-Level Markets

Georgios Tsaousoglou, Polyzois Soumplis, Nikolaos Efthymiopoulos, Konstantinos Steriotis, Aristotelis Kretsis, Prodromos Makris, Panagiotis Kokkinos, and Emmanouel Varvarigos

Abstract—The uncertain and non-dispatchable nature of renewable energy sources renders Demand Response (DR) a critical component of modern electricity distribution systems. Demand Response (DR) service provision takes place via aggregators and special distribution-level markets (e.g., flexibility markets), where small, distributed DR resources, such as building energy management systems, electric vehicle charging stations, micro-generation and storage, connected to the low-voltage distribution grid, offer DR services. In such systems, energy balancing (and thus, also DR decisions) have to be made close to real-time. Thus, market clearing algorithms for DR service provision must fulfill several requirements related to the efficiency of their operation. More specifically, a DR market clearing algorithm needs to be optimal in terms of cost-efficiency, scalable in terms of number of assets and locations, and able to satisfy real-time constraints. In order to cope with these challenges, this paper presents a distributed DR market clearing algorithm based on Lagrangian decomposition, combined with an optimal cloud resource allocation algorithm for assigning the required computation power. A heuristic algorithm is also presented, able to achieve a near-optimal solution, within negligible computational time. Simulations, performed on a testbed, demonstrate the computational burden introduced by various DR models, as well as the heuristic algorithm's near-optimal performance. The resource allocation algorithm is able to service multiple DR requests (e.g. in multiple distribution networks), and minimize the cost of computational resources while respecting the execution time constraints of each request. This enables third parties to offer cost-efficient and competitive DR operation as a service.

Index Terms—Smart Grid, Flexibility Markets, Demand Response, Cloud.

NOMENCLATURE

Sets and Indices

- N Set of DR facilities, indexed by n.
- T Set of timeslots in the scheduling horizon, indexed by t.
- Γ_n Set of flexibility assets of facility n, indexed by γ.
- $\mathcal{Y}_n$ Set of local control variables of facility n, indexed by y.
- C_n Set of local constraints of facility n.
- A Set of electricity grid nodes, indexed by a and i.
- B Set of electricity grid branches, indexed by ia.
- $\Omega_p(a)$ Set of parent nodes of node a.
- $\Omega_d(a)$ Set of children nodes of node a.
- N_a Set of facilities located at node a.
- k Index of algorithm iterations.
- R Set of DR requests, indexed by r.
- F_r Set of computational tasks (one task per facility f) of DR request r, indexed by f.
- V Set of nodes of the communication network, indexed by v or i, j.
- V_c Set of communication network nodes, with computational resources.
- V_f Set of communication network nodes, with DR facilities.
- E Set of virtual links, indexed by ij.
- M Set of types of computational resources, indexed by m.
- K_{ij} Set of shortest paths for ij, indexed by κ.

Variables

- $x_{n,t}$ Aggregated energy consumption of facility n at timeslot t.
- $p_{\gamma,t}$ Energy consumption of asset γ at timeslot t.
- $\theta_{a,t}$ RES curtailment factor for node a.
- $P_{ia,t}$ Active power flow between nodes i, a at t.
- $Q_{ia,t}$ Reactive power flow between nodes i, a at t.
- $V_{a,t}$ Voltage at node a, timeslot t.
- $\zeta_{v,m,f}$ Binary variable, for the assignment of the computational task f to a resource of type m at node v.
- $\beta_{v_f,v,\kappa}$ Binary variable, for a connection of nodes v and v_f (of task/facility f), through path κ.
- $\psi_{v,m,f}$ Starting timeslot of processing for task f, by a machine of type m, located at node v.
- $\xi_{v,m,f,\hat{f}}$ Binary variable, of whether task f is executed before $\hat{f}$ by m at v.

Parameters

- $\mathrm{P}^{\mathrm{d}}_{a,t}$ Power consumption at node a, timeslot t.
- $\mathrm{P}^{\mathrm{RES}}_{a,t}$ RES output at node a, timeslot t.
- $\mathrm{f}_{n,t}$ Parameter relating active and reactive power of facility n at t, through the power factor.

G. Tsaousoglou is with Eindhoven University of Technology and received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 754462. P. Soumplis, N. Efthymiopoulos, K. Steriotis, A. Kretsis, P. Makris, and E. Varvarigos are with the National Technical University of Athens. P. Kokkinos is with the Department of Digital Systems, University of the Peloponnese, Sparta, Greece. N. Efthymiopoulos, K. Steriotis, P. Makris, and E. Varvarigos received funding from European Union's Horizon 2020 research and innovation programme under grant agreement No. 863876 in the context of the FLEXGRID project.

R_{ia}	Resistance of line ia of the electricity grid.
X_{ia}	Reactance of line ia of the electricity grid.
$\mathrm{c}_a^{\mathrm{curt}}$	Per-unit cost of curtailing RES generation at node a.
π_t	Wholesale electricity price at timeslot t.
$\underline{\alpha_r}$	Arrival time of DR request r.
$\overline{\mathrm{lat}_r}$	Maximum acceptable latency for the service of DR request r.
$\delta_f^{\mathrm{in/out}}$	Size of input/output data of computational task f.
$\mathrm{c}_{v,m}^{\mathrm{proc}}$	Processing cost for machine type m, at node v.
$\mathrm{c}_{ij}^{\mathrm{net}}$	Network cost for virtual link ij.
tr_{ij}	Transmission rate of link ij.
$\mathrm{pr}_{v,m,f}$	Processing time of task f by machine type m at v.
$\mathrm{l}_{ij,\kappa}$	Length of shortest path κ for link ij.

1 Introduction

Towards facilitating the transition to a carbon-free electricity system, government policies introduce incentives for bottom-up investments in Renewable Energy Sources (RES). As a result, RES facilities are being installed in various locations of the medium or low-voltage distribution network. While these developments accelerate the penetration of RES, they do, however, create new challenges for power system operators. In particular, voltage and congestion issues become significant, as they can occur dynamically and close to real-time operation due to the volatile nature of RES output.

In order to avoid resorting to undesirable RES/load curtailments and costly grid reinforcement, Distribution System Operators (DSOs) can manage their networks and resolve voltage stability and congestion issues by drawing on the flexibility of small distributed resources located in the network [1], such as energy storage systems, micro-generation facilities and flexible loads, e.g., Heating Ventilation and Air Conditioning (HVAC), Electric Vehicles (EV), etc. We refer to these resources as flexibility assets, while the facility where they are located (building, charging station, etc.) is referred to as a flexible facility or simply facility.

Novel smart grid architectures (e.g. [2], [3]) have been proposed towards providing market frameworks for flexibility activation in such contexts. The relevant marketplaces are commonly referred to as "flexibility markets" [4], [5]. A flexibility market is the marketplace where a DSO dynamically procures Demand Response (DR) services [6] from assets located in different nodes of its network. In such markets, DR services are offered by flexible facility manager entities, who are responsible for aggregating, offering, and activating the facility's flexibility via DR actions. These DR actions are implemented by Energy Management Systems (EMS) that use Information-and-Communication Technology (ICT) to monitor and control the energy consumption of the flexible facility. Examples include Building EMS, EMS that monitor and control the charging power of electric vehicles in an EV charging station, and more.

The scalability properties of flexibility market-clearing algorithms constitute a critical issue towards bringing such solutions to real-life implementations. Moreover, each facility bears certain costs for performing DR actions, relating to the compensation (or energy bill discounts) that it should offer to its end users in order to modify the assets' energy consumption profile. The objective of the DSO is to satisfy the system's constraints in the most cost-efficient way, i.e., by drawing on the least expensive facility DR services. In addition, one needs to consider the minimization of the total system cost, including the DR procurement cost *and* the operational cost of the cloud services necessary to perform the various calculations required for the overall operation of the DR flexibility market.

Cloud computing applications for the smart grid architecture, e.g, [7] , [8] , [9], have mainly focused on three areas, namely, energy management, information management, and security. However, the need to support flexibility markets that consider physical network constraints of the distribution network through Alternate Current Optimal Power Flow (AC-OPF) formulations, has recently emerged [10]. The computational complexity, the robustness [11] and the scalability [12] of these solutions pose critical demands, requiring the efficient allocation of DR flexibility markets' computational tasks to computational resources so that the delay and processing requirements that these architectures need are guaranteed in an economically efficient manner. Moreover, as stated in many recent survey works, such as [13] and [14], traditional cloud computing architectures can hardly meet the requirements of large-scale real-time data processing in DR applications. Therefore, novel cloud-fog-edge computing architectures have been recently proposed, in which computation tasks can be decomposed and be allocated to edge/fog nodes and clouds through more effective task allocation strategies to strike an optimal trade-off between various requirements, such as computational complexity, scalability, time-related constraints and total operating costs.

Motivated by these developments, we propose, for the first time, an innovative business-to-business cloud service, noted as DR Operation as a Service (DROaaS), which facilitates DSOs, through the use of a DR-oriented dynamic cloud resource allocation framework. By exploiting intrinsic attributes of DR models to optimize cloud-based execution, the proposed framework provides the DSOs with a flexibility allocation algorithm that is: i) scalable in terms of number of assets and distribution network locations, ii) dynamic and able to make fast, real-time decisions, and iii) optimal (cost-efficient) in terms of minimizing the total system's cost (i.e., both DR procurement cost and cloud-related operational expenditures). The major contributions of this paper are summarized as follows:

A decomposition algorithm is presented, through which the flexibility market clearing problem is parallelized so that it becomes amenable to distributed (cloud) computation.

An integrated framework is developed that facilitates the realization of the DR operation as a service (DROaaS). This service calculates the optimal DR actions, while optimizing the cost of computational resources towards the proliferation of cost-competitive DR services.

The use of computational resources in the proposed multi-technology DR architecture is optimized through an integer linear programming algorithm. A heuristic algorithm is also presented, that achieves near-optimal performance with negligible computational time.

An extensive evaluation is performed, under a diverse set of end user devices and models. The evaluation results demonstrate the scalability, low delay and cost-competitiveness of the proposed architecture.

The remainder of the paper is structured as follows. The next subsection briefly surveys the most relevant works from the literature, emphasizing the differences with our proposed solution. Section 2 presents the architecture of the proposed DR system. Section 3 presents the modeling of a flexibility market, including an abstract model of DR resources and the DSO distribution network constraints, as well as a decomposition algorithm for tackling the optimal dispatch problem in a distributed fashion. Section 4 presents the optimal computational resource allocation algorithm, as well as a heuristic algorithm for making faster, near-optimal decisions. Section 5 presents the evaluation setup of the proposed DR architecture, including detailed models of DR resources and networks. Section 6 presents the simulation results, while Section 7 concludes this work.

Related Work

There are several works in the recent literature that propose a cloud-fog-edge architecture for dealing with DR operation. [15] presents a pioneering work in the exploitation of an Edge-Cloud architecture towards efficient DR in buildings. Additionally, in [16], the authors analyzed communication performance as a major requirement in cloud-based DR. More specifically, a cost-effectiveness analysis confirms that achieving higher performance incurs a higher communication cost. However, neither of the aforementioned works have dealt with the issue of adapting the allocation of DR computation tasks to computing resources. Consequently, there are no overall DR performance (delay and scalability) guarantees, while the satisfaction of the physical constraints of the power distribution network, and the consequent computational load it entails, are not considered.

The work in [17] presents an important effort on the use of clouds towards real time DR services, which is often referred to as Emergency Demand Response. The efficiency of the proposed solution is testified through performance evaluation results. In the same direction, [18] exploits clouds towards the real time management of smart grids. However, the former study does not consider the underlying distribution network, while the latter does not consider DR services.

Furthermore, [19] proposes an integration between smart grid and cloud (noted as Internet of Energy) by proposing a smart gateway that bridges the fog domain and the cloud. It is introduced for scheduling devices/appliances by creating a priority queue that can perform demand side management dynamically. However, [19] only presents a communication architecture and does not model the algorithmic problems of resource allocation and flexibility market-clearing.

The work in [20] proposes a cloud-edge cooperative control model and strategy for the price-based DR of large-scale Air Conditioners, while it is compared with a classic single cloud architecture model. The results show reduction of the grid's critical peak and elimination of the peak rebound. However, the computation tasks are statically allocated to the cloud or the edge, while our work uses a diverse set of DR assets, which incurs the need for dynamic allocation of computing resources.

The authors in [21] propose a 3-tier edge-cloud collaborative residential energy management architecture in order to alleviate fluctuations in demand, while reducing latency and improving processing performance. To this end, a two-level energy management mechanism was determined. The first stage models the interaction between real-time pricing and energy demand, while the second implements energy scheduling between the cloud tier, access tier, and infrastructure tier. [22] also proposes a similar 3-tier cloud-fog architecture that improves the response delay and uses a linearized AC-OPF model that finds the optimal solution. Edge computing resources are designed to generate Bender's cuts, and the cloud is designated as the coordinator of the whole process. Moreover, a few recent works, such as [23] and [24], deal with a distribution-level energy trading problem. [23] presents an energy trading management system, where the edge node acts as a retail energy market server providing energy services to the end-users. The architecture includes home gateways, local fog nodes and cloud server. The proposed edge/cloud model is compared to a classical single cloud-based one, showcasing its superiority with respect to network load and delay reduction. However, these works do not deal with optimal and dynamic allocation of computing resources and thus do not guarantee scalability and stringent delay constraints of the market clearing process.

A few more works, namely [25] and [26], consider cloud-edge architecture to deal with electric vehicle (EV) fleet management problem. In [25], cooperation among cloud and edge devices is realized to make intelligent decisions related to EVs' charging and discharging in addition to achieving the expected demand-supply balance,

without accounting for distribution network constraints. [26] transforms a traditional large-scale V2G problem into several sub-problems, which are small enough to optimize. Network constraints are also taken into account. In our work, we model diverse DR assets and not only EVs. We also propose an optimal solution for the orchestration of the heterogeneous cloud, fog and edge computing resources.

Finally, authors in [27] proposed the energy management as a service concept, which is implemented over a fog infrastructure. Scalability, adaptability, delay constraints and cloud cost minimization are some of the requirements that are extensively discussed. However, this is rather a high level analysis, which means that there is neither a mathematical problem formulation nor a proposed algorithm included. In contrast, our work co-optimizes the cost of DR procurement and cloud resources by developing a solid mathematical model and algorithmic solution to realize the novel DROaaS business model.

2 System Architecture

The proposed DR architecture assumes a computing and networking (COMNET) infrastructure that interacts with the EMS and supports the operation of the DR mechanisms. The COMNET infrastructure combines heterogeneous resources from the edge/fog layers to bring adequate resources close to flexibility assets, and from multiple clouds (federated operation).

Computing resources can range from generic ones, to specialized computing devices (FPGA, GPU), to micro-DCs and larger DCs, deployed in urban (office and residential buildings) and rural areas (e.g., alongside farms of wind turbines and solar panels), some closer to the edge and some deeper in the cloud forming the edge-fog-cloud hierarchy. Moreover, these resources may belong to different administrative authorities (providers) thus forming a hierarchy of privately owned and public computational resources. Moving from the lower layers of the hierarchy to the higher ones, the provided capacity, scalability and resiliency increase, but so does the delay. Edge resources can perform light computations and filtering functions, while complex computations have to be offloaded to the higher layers, i.e., deeper in the cloud. Such approaches are currently being adopted in other time-critical applications, e.g. closed-circuit television cameras fitted with artificial intelligence capabilities for facial recognition technology. We regard that the same approach is relevant for the smart energy field. More specifically, we consider as edge resources the resources that operate within and/or close to each facility featuring low network delay but low computational capacity. We also note as fog resources the resources located on the local DSO data center (i.e. dedicated servers, which are available to compute purpose-specific applications). Finally, we note as cloud resources the ones located at large data centers and are typically owned by large cloud service providers (e.g., Amazon, Google, etc). These usually have larger network delay but higher capacity.

The networking infrastructure includes various networking mechanisms using different wired (optical) and wireless (e.g. 5G) technologies to provide the required interconnection of the computing resources over private and public network infrastructures. These multi-domain and multi-technology network paths are controlled and managed by the telco operators based on Software Defined Networking (SDN) principles. Hence, we abstract the communication paths between the resources in the same or different layers as virtual links with specific latency and capacity. These values depend on the networking locality of the resources, with those in proximity resulting in lower latency than those that are far apart.

Each facility's EMS infrastructure contains sensors, actuators and/or smart plugs, together with appropriate interfaces through which end users are able to set their preferences regarding the use of their flexibility assets for providing DR. By drawing on the EMS monitoring and control capabilities, the flexible facility can offer DR services to the DSO. In turn, the DSO needs to decide the optimal configuration of DR-services (e.g., which facilities should activate their flexibility and by how much). This optimization can be mathematically decomposed into smaller subproblems (computational tasks), which can be performed in a cost-effective and time-critical manner by drawing on the COMNET infrastructure. Thus, a business model is enabled, where a third party can offer a DR operation as a service (DROaaS) to multiple systems of DSOs and flexibility asset owners/users.

Fig. 1 depicts the proposed DR Operation as a Service that can orchestrate the decomposed instances of the market clearing algorithm over the available COMNET infrastructure. The main components that form the proposed DROaaS architecture are:

An EMS per facility that exploits ICT technology to:
- monitor and control the flexibility assets of that facility,
- allow end-users to declare their electricity consumption preferences through a user interface, and
- communicate the facility's DR capabilities and receive dispatch orders.

The Service Orchestrator provides the necessary interface between the DROaaS platform and the facilities and DSOs. It receives, through its interface, the requirements and specifications of the DR market clearing problem (Sections 3.1-3.2).

The Resource Orchestrator decomposes the DR market clearing problem into smaller subproblems (Section 3.3), and assigns the subtasks to the most appropriate edge, fog, or cloud computational resources (Section 4).

The Infrastructure Manager handles the interaction with the local orchestrators at the various com-

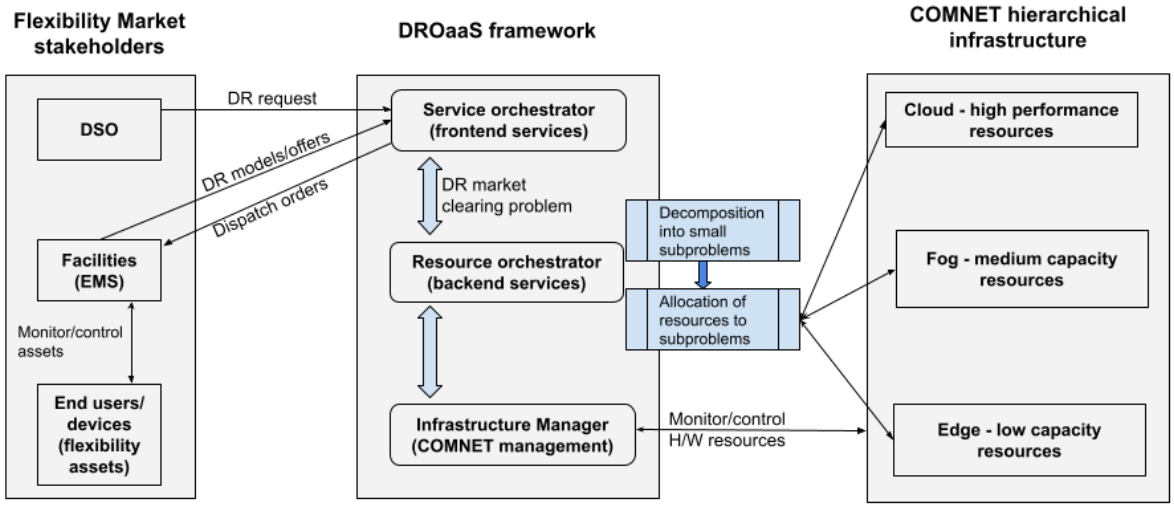

Fig. 1. DR Operation as a service

putational resources, models their capabilities and enables their monitoring.

In the following Section, we elaborate on the problem definition and decomposition, while in Section 4 we present the algorithms for the allocation of the decomposed subproblems to computational resources.

3 DR Problem Formulation and Decomposition

3.1 System Model

A flexibility market consists of a set $N = \{0, 1, 2, ..., |N|\}$ of flexible facilities (e.g., buildings, EV charging stations, storage facilities, etc) and a DSO, noted as participant 0 of set N. Each facility $n \in N/\{0\}$ is located at a particular node of the DSO's distribution network and can perform DR actions. Continuous time is divided into a set T of timeslots for an horizon ahead.

Each facility is able to control the power consumption of its flexibility assets through the facility's EMS. The set of flexibility assets of facility n is denoted by Γ_n. The aggregated energy consumption of a facility at timeslot t is denoted by $x_{n,t}$, while the energy consumption of a particular asset $\gamma \in \Gamma_n$ of facility n, is denoted by $p_{\gamma,t}$. Therefore, we have

$$\sum_{\gamma \in \Gamma_n} p_{\gamma,t} = x_{n,t}, \quad \forall n \in N/\{0\}, t \in T. \tag{1}$$

Each facility features a set $\mathcal{Y}_n$ of local variables, which includes $x_{n,t}, p_{\gamma,t}$ (for every t and γ) and also other local variables, depending on the particular models of the facility's flexibility assets. A facility also bears a set C_n of feasible operational points, defined by a number of operational constraints on the combinations $\{y\}_{y \in \mathcal{Y}_n}$ of all local variables $y \in \mathcal{Y}_n$. Therefore, we have

$$\{y\}_{y \in \mathcal{Y}_n} \in C_n, \quad \forall n \in N/\{0\}. \tag{2}$$

Detailed asset models are presented in Section 5. While those models facilitate the adequate evaluation of the proposed architecture, the architecture is open and transparent to the facility DR models used, in the sense that it is not bounded to those, or any other, particular models. For this reason, the operational constraints (2) are kept in an abstract and general form for now.

The DSO is responsible for maintaining the distribution network within safe operational limits. Assuming a radial network, let A denote the set of network nodes and B the set of branches. For a node $a \in A$, let $\Omega_p(a)$ (or $\Omega_d(a)$) denote the set of predecessor (or descendant, respectively) nodes connected to node a. In node a, there is a certain amount of power consumption $\mathrm{P}^{\mathrm{d}}_{a,t}$, as well as a RES power generation $\mathrm{P}^{\mathrm{RES}}_{a,t}$, which the DSO can choose to curtail by a factor of $1 - \theta_{a,t} \in [0, 1]$, where $\theta_{a,t} = 1$ means that there is no RES curtailment and $\theta_{a,t} = 0$ means that the whole RES output of node a is curtailed. Finally, let N_a denote the set of facilities located at node $a \in A$. Towards modeling the flows and constraints of the physical electricity grid, we use the linearized DistFlow equations [28], defined by the following set of constraints:

$$\sum_{i \in \Omega_p(a)} P_{ia,t} - \sum_{n \in N_a} x_{n,t} - \mathrm{P}^{\mathrm{d}}_{a,t} + \theta_{a,t}\mathrm{P}^{\mathrm{RES}}_{a,t} = \sum_{j \in \Omega_d(a)} P_{aj,t}, \quad \forall a \in A, t \in T \tag{3}$$

$$\sum_{i \in \Omega_p(a)} Q_{ia,t} - \sum_{n \in N_a} \mathrm{f}_{n,t} x_{n,t} - \mathrm{Q}^{\mathrm{d}}_{a,t} + \theta_{a,t}\mathrm{f}^{\mathrm{RES}}_{a,t}\mathrm{P}^{\mathrm{RES}}_{a,t} = \sum_{j \in \Omega_d(a)} Q_{aj,t}, \quad \forall a \in A, t \in T \tag{4}$$

$$V_{a,t} = V_{i,t} - 2(\mathrm{R}_{ia} P_{ia,t} + \mathrm{X}_{ia} Q_{ia,t}), \quad \forall a \in A, i \in \Omega_p(a), t \in T \tag{5}$$

$$\underline{\mathrm{V}}_a \leq V_{a,t} \leq \overline{\mathrm{V}}_a, \quad \forall a \in A, t \in T \tag{6}$$

$$\underline{\mathrm{P}}_{aj,t} \leq P_{aj,t} \leq \overline{\mathrm{P}}_{aj,t}, \qquad \forall aj \in B, t \in T \quad (7)$$

$$\underline{\mathrm{Q}}_{aj,t} \leq Q_{aj,t} \leq \overline{\mathrm{Q}}_{aj,t}, \qquad \forall aj \in B, t \in T, \quad (8)$$

where $\mathrm{f}_{n,t}$ are the parameters relating active and reactive power (through the power factor), and $P_{ia,t}$ and $Q_{ia,t}$ are the active and reactive power flowing on the branch ia connecting nodes $i \in A$ and $a \in A$ of the distribution network.

Eq. (3) and Eq. (4) represent the active and reactive power balances at each distribution node. Namely, the total outgoing power $\sum_{j \in \Omega_d(a)} P_{aj,t}$ from node a, equals the incoming power $\sum_{i \in \Omega_p(a)} P_{ia,t}$ minus the net power consumption $\sum_{n \in N_a} x_{n,t} + \mathrm{P}^{\mathrm{d}}_{a,t} - \theta_{a,t}\mathrm{P}^{\mathrm{RES}}_{a,t}$ of node a. Eq. (5) describes the voltage drop between each pair of neighboring nodes a, i where $i \in \Omega_p(a)$. Variable $V_{a,t}$ denotes the squared voltage of node a at t, while R_{ia} and X_{ia} are the resistance and reactance, respectively, of branch ia. The grid's voltages and active/reactive power flows must satisfy certain limits to ensure the physical grid's operational safety. Constraints (6) make sure that voltages in all nodes stay within safe margins, while (7) and (8) limit the active and reactive power flows for all branches.

3.2 DR Problem Formulation

The DSO decides the amount of RES curtailments $\theta_{a,t}$, that come at a cost of $\mathrm{c}^{\mathrm{curt}}_a$ per 1 MW of RES generation curtailment, as well as variables $x_{0,t}$ that express the power exchange with the main grid. A cost function $d_0(\mathcal{Y}_0)$ is defined for the DSO, to capture the cost of exchanging energy with the main grid and the cost of RES curtailments, namely,

$$d_0(\mathcal{Y}_0) = \left(\sum_{t \in T} \pi_t x_{0,t} + \sum_{a \in A} (1 - \theta_{a,t})\mathrm{P}^{\mathrm{RES}}_{a,t}\mathrm{c}^{\mathrm{curt}}_a \right), \quad (9)$$

where π_t is the price for importing/exporting energy and $\mathcal{Y}_0 = \{P_{ia}, P_{aj}, Q_{ia}, Q_{aj}, V_{a,t}, x_{0,t}, \theta_{a,t}\}$.

On the other hand, each facility bears a DR-cost function $d_n(\mathcal{Y}_n)$, where $\mathcal{Y}_n$ denotes the set of local variables of the facility. The function $d_n(\cdot)$ is used by the facility to model the DR costs of its assets. For example, an EV charging station would need to compensate its EV users (or offer price discounts) to counteract their dissatisfaction for suffering delays in their battery charging due to congestion in the electricity network. Similarly to constraints (2), facility DR-cost functions are kept in a general form in this Section, but they are modeled explicitly in Section 5 for the evaluation tests.

The objective of the market clearing algorithm is to make sure that the network operates within the feasible operational area, while minimizing the aggregate system cost (i.e., the cost of DR actions and the cost of exchanging power with the main grid), as in

$$\min \left(\sum_{n \in N} d_n(\mathcal{Y}_n) \right) \quad \text{s.t.} (1) - (9). \quad (10)$$

The intuition behind problem (10) is that, in case the satisfaction of the physical grid's safety constraints necessitates DR actions, RES curtailments and/or power imports, the DSO will decide the least expensive combination of actions. For example, facilities that have a very small DR cost (e.g., a battery or a very flexible load) will be prioritized for dispatch actions before modifying the consumption profile of critical loads or resorting to RES curtailments and/or power imports at times where the electricity prices are high.

3.3 DR Problem Decomposition

Solving problem (10) directly, in a centralized fashion, poses a number of challenges. The first challenge is that all models (DR cost functions and operational constraints) of the facilities would need to be communicated to a central entity, which raises security and privacy concerns. A second issue is that the large number of variables makes the problem computationally intensive.

In order to overcome these issues, problem (10) can be solved in a distributed fashion using a Lagrangian decomposition. Each facility solves a local optimization problem to decide the value of its local variables $\mathcal{Y}_n$, while the DSO solves an optimal power flow problem. The procedure iterates, while coordination is achieved by updating a set of Lagrange multipliers $\lambda_{a,t}$ and $\mu_{a,t}$ that are related to the dual variables of the active and reactive power balance constraints, respectively. More specifically, we consider the alternating direction method of multipliers (ADMM). By taking the augmented Lagrangian of problem (10), we have

$$\mathcal{L} = \sum_{n \in N} d_n(\mathcal{Y}_n) + \sum_{a \in A} \sum_{t \in T} \lambda_{a,t} g_{a,t} + \frac{\rho_1}{2}\|g_{a,t}\|^2 + \sum_{a \in A} \sum_{t \in T} \mu_{a,t} h_{a,t} + \frac{\rho_2}{2}\|h_{a,t}\|^2 \quad (11)$$

where

$$g_{a,t} = \sum_{i \in \Omega_p(a)} P_{ia,t} + \sum_{n \in N_a} x_{n,t} + \mathrm{P}^{\mathrm{d}}_{a,t} - \theta_{a,t}\mathrm{P}^{\mathrm{RES}}_{a,t} - \sum_{j \in \Omega_d(a)} P_{aj,t} \quad (12)$$

$$h_{a,t} = \sum_{i \in \Omega_p(a)} Q_{ia,t} + \sum_{n \in N_a} \mathrm{f}_{n,t} x_{n,t} + \mathrm{Q}^{\mathrm{d}}_{a,t} - \theta_{a,t}\mathrm{f}_{a,t}\mathrm{P}^{\mathrm{RES}}_{a,t} - \sum_{j \in \Omega_d(a)} Q_{aj,t}. \quad (13)$$

An iterative method for solving problem (10) is defined based on the following variable update rules

Facility

$$\{y\}^{(k+1)}_{y \in Y_n} = \underset{Y_n}{\operatorname{argmin}} \left\{ \mathcal{L}^{(k)} \right\} \quad \text{s.t.} (1), (2) \quad (14)$$

Algorithm 1: Iterative distributed algorithm for solving problem (10)

Initialize $k = 0, \lambda_{a,t}^{(0)} = 0, \mu_{a,t}^{(0)} = 0$;
while $g_{a,t}, h_{a,t} \geq \varepsilon$ **do**
  Multipliers $\lambda_{a,t}^{(k)}, \mu_{a,t}^{(k)}$ are communicated to all computational resources;
  The resource responsible for facility n solves problem (14);
  The solutions are communicated to the computational resource responsible for the resource responsible for the DSO, which solves (15);
  The solutions are communicated to the computational resource responsible for the multiplier and iteration updates;
  Multipliers are updated based on (16), (17) and $k = k + 1$.

DSO

$$P_{ia}^{(k+1)}, P_{aj}^{(k+1)}, Q_{ia}^{(k+1)}, Q_{aj}^{(k+1)}, V_{a,t}^{(k+1)}, \theta_{a,t}^{(k+1)} = \text{argmin}_{P_{ia}, P_{aj}, Q_{ia}, Q_{aj}, V_{a,t}, \theta_{a,t}} \mathcal{L}^{(k+1)} \quad \text{s.t. } (5) - (9) \tag{15}$$

Coordinating Entity

$$\lambda_{a,t}^{(k+1)} = \lambda_{a,t}^{(k)} + \rho_1 g_{a,t}^{(k+1)} \tag{16}$$

$$\mu_{a,t}^{(k+1)} = \mu_{a,t}^{(k)} + \rho_2 h_{a,t}^{(k+1)} \tag{17}$$

where ρ_1 and ρ_2 are step update coefficients.

This formulation allows problem (10) to be parallelized in order to be solved by appropriate computational resources in a coordinated distributed fashion. In particular, the computing task of each facility, i.e. solving problem (14), can be viewed as a self-dispatch problem where the facility decides the power consumption of each asset under the current active and reactive electricity prices $\lambda_{a,t}^{(k)}, \mu_{a,t}^{(k)}$ of the facility's node. On the other hand, the DSO solves an optimal power flow problem, i.e. (15), to decide whether any RES generation needs to be curtailed as well as the amount of power exchange with the main grid. The sequence and variable exchange among the execution nodes that execute each function is described in Algorithm 1.

4 Cloud Resource Allocation

We consider the functionality of DROaaS, as a means to efficiently coordinate the calculations of a set R of DR requests (corresponding to different distribution networks), where a DR request $r \in R$ is an instance of problem (10) that is solved through Algorithm 1. Each particular DR request is decomposed into $N + 1$ subproblems, as presented in the previous section. Each subproblem corresponds to the local optimization problem of a facility (or the DSO) and can be viewed as a different computation task. Therefore, each DR request $r \in R$ is characterized by its arrival time α_r, its set of tasks F_r (facilities and DSO) and an upper bound $\overline{\text{lat}}_r$ on the allowable latency per iteration.

Each task requires the transmission of input data δ_f^{in} and output data δ_f^{out}. These tasks can be executed in the facilities (edge resources) that they originate from, or be forwarded to aggregation points (fog resources) or to the cloud, and they introduce a latency lat_f. The incentive for moving tasks from the edge (on site) to the fog (other sites) and to the cloud (central sites) is the lower capacity of lower-level resources that may prolong the execution time of the bag of tasks beyond the allowable delay. Aggregating multiple tasks in fog resources can reduce the overall cost of the operation, assuming that resources' marginal cost reduces as a function of the submitted workload. On the other hand, moving tasks to higher layers of the COMNET infrastructure introduces networking latency that may increase the tasks' overall execution time, the so-called makespan.

Let graph $G = (V, E)$ jointly represent the flexibility markets and the computing and networking (COMNET) infrastructure. The set $V = V_c \cup V_f$ consists of the nodes V_c that possess computational resources, and the nodes $V_f : f \in F_r,\ r \in R$ where the facilities are connected. Facility nodes can be also equipped with processing units, thus $V_c \cap V_f \neq \varnothing$ in general. The set E corresponds to virtual links that interconnect the nodes over wired and wireless communication paths. Let M denote the set of types of computational resources available in the system. A resource of type $m \in M$, located at node $v \in V$, is characterized by a processing cost $\text{c}_{v,m}^{\text{proc}}$. Each virtual link $i, j \in E$ is characterized by a network cost $\text{c}_{i,j}^{\text{net}}$, depending on its available networking capacity. The overall transmission rate of the virtual link i, j is denoted as $\text{tr}_{i,j}$, resulting in a transmission latency for the data that needs to be transferred between nodes i and j and a propagation latency that depends on the physical distance of the virtual link.

The goal of the resource optimization procedure is to minimize the weighted sum of the processing per iteration cost and the latency for serving all the DR requests in R, while respecting the time constraints of each request. In the next subsections we formulate the problem as an ILP, and also provide a heuristic algorithm for keeping the computational time low.

4.1 Optimal ILP for the Allocation of Computational Resources

In what follows, we present the ILP formulation of the dynamic resource allocation problem. We use the index f to refer to a facility (and respective computation task) of any DR request, i.e. $f \in F$, where $F = \bigcup_{r \in R} \{F_r\}$. Let binary variable $\zeta_{v,m,f}$ denote whether a virtual machine of type m located at node v, is assigned to perform the

calculation task f. A resource m at node v needs $\mathrm{pr}_{v,m,f}$ time to perform the computations of task f.

To speed-up the calculations we make use of a pre-processing phase in which we pre-calculate κ shortest paths with length $\mathrm{l}_{i,j,\kappa}$ between each pair of nodes $i,j \in E$, which include the paths from the location v_f of each facility f, to the different processing nodes $v \in V$ and between the processing nodes. Given the communication network topology G, let $K_{i,j}$ denote the set of κ shortest paths between nodes i,j, and set $\Lambda_{i,j}$ contain their respective lengths $\mathrm{l}_{i,j,\kappa}$. Then, binary variable $\beta_{v_f,v,\kappa}$ denotes whether the corresponding facility of task f (located at node v_f) is connected to a node v over virtual link $\kappa \in K_{v_f,v}$ or not.

An integer variable, denoted as $\psi_{v,m,f}$, indicates the timeslot[1] in which a resource of type m, located at node v, starts the processing of task f. Finally, binary variable $\xi_{v,m,f,\mathfrak{p}}$ denotes whether the calculation of task f at m,v is performed before that of task $\mathfrak{p}$. The objective of optimal resource allocation is to minimize the overall processing and network costs, i.e.

$$\min_{\mathcal{W}} \left\{ \mathrm{w} \cdot \sum_{v \in V} \sum_{m \in M} \sum_{f \in F} \zeta_{v,m,f} \cdot \mathrm{c}^{\mathrm{proc}}_{v,m} + (1-\mathrm{w}) \cdot \sum_{v \in V} \sum_{m \in M} \sum_{f \in F} (\psi_{v,m,f} + \zeta_{v,m,f} \cdot \mathrm{pr}_{v,m,f}) \right\}, \tag{18}$$

where $\mathcal{W} = \{\zeta_{v,m,f}, \beta_{v_f,v,\kappa}, \psi_{v,m,f}, \xi_{v,m,f,\mathfrak{p}}\}$ and w is an objective weighting coefficient taking values between 0 and 1. When $\mathrm{w} = 0$ the latency for serving the DR requests is minimized, while when $\mathrm{w} = 1$, the processing per iteration cost is minimized. In intermediary cases where cost and latency are traded off, the value of w needs to be appropriately tuned so that the processing cost (measured in monetary units) is balanced with the latency value (measured in units of time). The optimization is subject to the following constraints. Each task has to be assigned to exactly one virtual machine:

$$\sum_{v \in V} \sum_{m \in M} \zeta_{v,m,f} = 1, \ \ \forall f \in F. \tag{19}$$

In order to assign task f to resource v,m, a connection path must be selected:

$$\sum_{\kappa \in K_{v_f,v}} \beta_{v_f,v,\kappa} \geq \sum_{m \in M} \zeta_{v,m,f}, \ \ \forall f \in F, v \in V. \tag{20}$$

We assume that the multiplier updates are made by the DSO itself[2]. Let $\mathbf{e}_r$ denote the node where the DSO of DR request r is located and $\mathfrak{e}_r$ denote the special task of multiplier update. Each facility allocates a virtual link for forwarding the data to the DSO

$$\sum_{\kappa \in K_{v_f,\mathbf{e}_r}} \beta_{v_f,\mathbf{e}_r,\kappa} = 1, \ \ \forall f \in F_r, r \in R. \tag{21}$$

Node v cannot begin the execution of task f before receiving the input data δ^{in}_f of f. This is subject to transmission and propagation delays, and the starting time of task f's at m,v can be calculated to be

$$\psi_{v,m,f} \geq \frac{\beta_{v_f,v,\kappa}\mathrm{l}_{v_f,v,\kappa}}{\Phi} + \frac{\delta^{\mathrm{in}}_f}{\mathrm{tr}_{v_f,v}} - (1 - \sum_{m \in M} \zeta_{v,m,f}) \cdot \mathrm{Q}, \quad \forall m \in M, v \in V, f \in F, \tag{22}$$

where Φ is the speed of light and Q is a sufficiently big number. The DSO cannot update the multipliers before receiving the response of each calculation task, implying that

$$\psi_{\mathbf{e}_r,m,\mathfrak{e}_r} \geq \psi_{v,m,f} + \mathrm{pr}_{v,m,f} + \frac{\beta_{\mathbf{e}_r,v_f,\kappa}\mathrm{l}_{\mathbf{e}_r,v_f,\kappa}}{\Phi} + \frac{\delta^{\mathrm{out}}_f}{\mathrm{tr}_{\mathbf{e}_r,v_f}} - (1 - \sum_{m \in M} \zeta_{\mathbf{e}_r,m,\mathfrak{e}_r}) \cdot \mathrm{Q}, \quad \forall m \in M, v \in V, f \in F_r, r \in R. \tag{23}$$

When the multipliers are updated, an iteration of the distributed algorithm is completed and the respective latency per iteration constraint must be satisfied:

$$\psi_{\mathbf{e}_r,m,\mathfrak{e}_r} + \mathrm{pr}_{\mathbf{e}_r,m,\mathfrak{e}_r} \leq \overline{\mathrm{lat}}_r, \quad \forall m \in M, v \in V, f \in F_r, r \in R. \tag{24}$$

Finally, the following three constraints ensure that the execution time ordering is preserved and there are no overlaps (i.e., simultaneous task executions at the same machine):

$$\xi_{v,m,f,\mathfrak{p}} + \xi_{v,m,\mathfrak{p},f} = 1, \quad \forall m \in M, v \in V, f, \mathfrak{p} \in F, f \neq \mathfrak{p} \tag{25}$$

$$\psi_{m,v,f} + \mathrm{pr}_{v,m,f} - \psi_{m,v,\mathfrak{p}} \leq 1 - \xi_{v,m,f,\mathfrak{p}} + 2 - \zeta_{v,m,f} - \zeta_{v,m,\mathfrak{p}} \ \cdot \mathrm{Q} \quad \forall m \in M, v \in V, f, \mathfrak{p} \in F, f \neq \mathfrak{p} \tag{26}$$

$$\psi_{m,v,\mathfrak{p}} + \mathrm{pr}_{m,v,\mathfrak{p}} - \psi_{m,v,f} \leq 1 - \xi_{v,m,\mathfrak{p},f} + 2 - \zeta_{v,m,\mathfrak{p}} - \zeta_{v,m,f} \ \cdot \mathrm{Q} \quad \forall m \in M, v \in V, f, \mathfrak{p} \in F, f \neq \mathfrak{p}. \tag{27}$$

For large instances, the optimal ILP solution can take a long time to calculate. Thus, in the next subsection, we present a heuristic algorithm that can achieve a near-optimal solution with minimal computational time.

1. The set T of timeslots defined in Section 3 refers to operational timeslots e.g. of 15-minute duration. On the contrary, here we refer to timeslots that relate to the execution times of the calculations. Those are of much smaller durations. In fact, these timeslots belong to a set T, where the total duration of all timeslots in T, is smaller than the duration of one timeslot $t \in T$, in order to satisfy the requirement that the calculations' execution should finish before the operational timeslot changes.

2. This is a plausible assumption since the DSO is responsible for coordinating the dispatch actions of its distribution network. However, the assumption is without loss of generality, since a third party could also be responsible for the simple operation of multipliers' update.

Algorithm 2: DROaaS heuristic algorithm

Input: G=(V,E), M, R, w, $\{\mathrm{c}^{\mathrm{proc}}_{v,m}\}_{\forall v \in V, m \in M}$, $\{\mathrm{c}^{\mathrm{net}}_{i,j}\}_{\forall (i,j) \in E}$

Output: Allocation of facility tasks to computational resources: $\{\zeta_{v,m,f}, \psi_{v,m,f}\}_{\forall v,m,f}$

Initialize: $\{\zeta_{v,m,f}, \psi_{v,m,f}\}_{\forall v,m,f} = 0$

1 Sort the DR requests in descending order of their latency bound $\overline{\mathrm{lat}}_r$;

2 **for** *each DR request* $r \in R$ **do**

3 **for** *each facility task* $f \in F_r$ **do**

4 Update $\zeta_{v,m,f}, \psi_{v,m,f}$, by solving problem (18)-(27) with $\{\zeta_{v,m,f'}, \psi_{v,m,f'}\}_{\forall v,m,f' \neq f}$ fixed, and with constraint (23) relaxed

end

5 Update $\zeta_{v,m,f'}, \psi_{v,m,f'}$, by solving problem (18)-(27) to allocate the DSO tasks, with the allocation $\{\zeta_{v,m,f}, \psi_{v,m,f}\}_{\forall v,m,f: f \neq f'}$ of the facilities' tasks fixed

end

4.2 Heuristic Algorithm for Fast Resource Allocation

The heuristic algorithm decomposes the selection of processing nodes and transmission links in a separable form, assuming there is no coupling between the nodes where the processing is performed and the node which is responsible for communicating the multipliers and the iteration update. Thus, the problem can be efficiently solved by solving the two decoupled problems, where the first finds the pairing of facility tasks to processing nodes, and the second finds the pairings of facility nodes to the node responsible for the iterations. Since the number of variables and constraints in this case is not large, many algorithms can be applied. The heuristic, presented in Algorithm 2, is based on relaxing the grouping constraints by first assigning facility tasks to processing nodes and then DR requests to the communicating node.

More specifically, the heuristic algorithm serves the DR requests sequentially, one by one. To do so, the DR requests are sorted in descending order based on their service latency requirements (line 1). Hence, decisions for the DR requests with strict latency requirements are prioritized. Then, the facility tasks of the request are examined sequentially, and resources are allocated based on a best fit approach and the selected objective function (lines 3-4). The allocation of each facility task is determined (and updated) by solving problem (18) - (27), but keeping all the variables of other tasks fixed. Since the DSO's task has not been allocated at this point (i.e., the DSO variables are initialized to zero), constraint (23) is relaxed to prevent infeasibility. After all facility tasks are allocated, the algorithm allocates resources to the DSO task (line 5), by solving (18) - (27), while keeping the variables of facility tasks fixed.

5 Evaluation Setup

In this section we present the evaluation setup, which includes a set of detailed heterogeneous facility DR models, a benchmark DSO network, and COMNET infrastructure.

5.1 Facility DR models

We consider several heterogeneous facility DR models, where the differences lie in the modeling choices of the facility manager entity or in the nature of the facility's flexible loads. All flexible loads are characterized by minimum and maximum operational points between which the load's electricity consumption must lie, i.e.,

$$\underline{\mathrm{x}}_{\gamma} \leq x_{\gamma,t} \leq \overline{\mathrm{x}}_{\gamma}, \quad \forall \gamma \in \Gamma_n, n \in N \tag{28}$$

The DR cost d_n of facility n is defined as the sum of the DR costs $d_{\gamma,n}$ of all the assets that it operates, i.e.

$$d_n(\mathcal{Y}_n) = \sum_{\gamma \in \Gamma_n} d_{\gamma,n}(\mathcal{Y}_n), \quad \forall n \in N. \tag{29}$$

In the following subsections, we present the facility DR models used in the simulations. A facility DR model refers to the specific formulation of the facility's constraints (2) and its DR cost function $d_n(\mathcal{Y}_n)$. Based on its DR model, each facility type bears different computational requirements for solving its local problem (14), which greatly interferes with the resource allocation problem. The particular values for all parameters used in the following subsections are presented in Table 1.

5.1.1 Facility with curtailable loads

The set of facilities belonging to this type is denoted by N_{curt}. A curtailable load $\gamma \in \Gamma_{n: n \in N_{\mathrm{curt}}}$ has a desired consumption $\mathrm{e}_{\gamma,t}$ at timeslot t and is characterized by a set of DR cost parameters $\mathrm{c}_{\gamma,t}$ that relate to the level of the load's inelasticity. For these loads the set of controllable variables consists only of the electricity consumption variables $x_{\gamma,t}$, i.e. $\mathcal{Y}_{n: n \in N_{\mathrm{curt}}} = \{x_{\gamma,t}\}_{\gamma \in \Gamma_n, t \in T}$. The DR cost function of a load, as adapted by [29] and [30], is defined by

$$d_{\gamma,n}(\mathcal{Y}_n) = \sum_{t \in T} d_{\gamma,n,t}, \quad \forall \gamma \in \Gamma_n, n \in N_{\mathrm{curt}}, \tag{30}$$

where

$$d_{\gamma,n,t} = \mathrm{c}_{\gamma,t}(\mathrm{e}_{\gamma,t} - x_{\gamma,t})^2, \quad \forall \gamma \in \Gamma_n, n \in N_{\mathrm{curt}}. \tag{31}$$

5.1.2 Facility with curtailable loads and ramp constraints

For some assets it might be relevant to constraint the ramp up/down rates $\mathrm{r}^{\mathrm{up}}/\mathrm{r}^{\mathrm{down}}$ of energy consumption, in order to avoid abrupt changes in their consumption from one timeslot to the next. For this type of loads $\gamma \in \Gamma_n, n \in N_{\mathrm{ramp}}$, the control variables, constraints and cost function are the same as those of curtailable loads, but with an additional time-coupling constraint:

$$\mathrm{r}^{\mathrm{down}}_{\gamma} \leq x_{\gamma,t} - x_{\gamma,t-1} \leq \mathrm{r}^{\mathrm{up}}_{\gamma}, \quad \forall \gamma \in \Gamma_n, n \in N_{\mathrm{ramp}}. \tag{32}$$

5.1.3 Facility with time-shiftable loads

This set of facilities is denoted by N_{shift}. A load $\gamma \in \Gamma_{n:n2N_{\text{shift}}}$ has a desired energy consumption E_γ that must be fulfilled within the time interval $[\mathrm{t}_\gamma^{\text{arr}}, \mathrm{t}_\gamma^{\text{dep}}]$:

$$\sum_{t2[\mathrm{t}_\gamma^{\text{arr}},\mathrm{t}_\gamma^{\text{dep}}]} x_{\gamma,t} = \mathrm{E}_\gamma, \quad \forall \gamma \in \Gamma_n, n \in N_{\text{shift}} \tag{33}$$

The only decision variables are again $x_{\gamma,t}$. The load also has a desired completion time $\hat{\mathrm{t}}_\gamma \leq \mathrm{t}_\gamma^{\text{dep}}$. If part of the load's required energy consumption is consumed after $\hat{\mathrm{t}}_\gamma$, then the load bears a cost, defined as

$$d_{\gamma,n}(\mathcal{Y}_n) = \sum_{t2T} d_{\gamma,n,t}, \quad \forall \gamma \in \Gamma_n, n \in N_{\text{shift}}, \tag{34}$$

where

$$d_{\gamma,n,t} = \sum_{t=\hat{\mathrm{t}}_\gamma+1}^{\mathrm{t}_\gamma^{\text{dep}}} \frac{\mathrm{s}_\gamma^{(t-\hat{\mathrm{t}}_\gamma)}}{\mathrm{E}_\gamma} x_{\gamma,t}, \quad \gamma \in \Gamma_n, \forall n \in N_{\text{shift}} \tag{35}$$

Intuitively, the term $\mathrm{s}_\gamma^{(t-\hat{\mathrm{t}}_\gamma)}$ imposes a higher DR cost for later timeslots through the exponent, while parameter $\mathrm{s}_\gamma \geq 1$ relates to the load's inelasticity. This model is adapted from [31].

5.1.4 Facility with fully flexible loads

This set of facilities is denoted by N_{flex}. A load $\gamma \in \Gamma_{n:n2N_{\text{flex}}}$ is characterized by a feasible time interval $[\mathrm{t}_\gamma^{\text{arr}}, \mathrm{t}_\gamma^{\text{dep}}]$, as well as a desired energy consumption $\hat{\mathrm{E}}_\gamma$ and a minimum acceptable energy consumption $\underline{\mathrm{E}}_\gamma$ for that time interval, i.e.

$$\underline{\mathrm{E}}_\gamma \leq \sum_{t2[\mathrm{t}_\gamma^{\text{arr}},\mathrm{t}_\gamma^{\text{dep}}]} x_{\gamma,t} \leq \hat{\mathrm{E}}_\gamma, \quad \forall \gamma \in \Gamma_n, n \in N_{\text{flex}}. \tag{36}$$

The set of controllable variables is again $\mathcal{Y}_{n:n2N_{\text{flex}}} = \{x_{\gamma,t}\}_{\gamma 2 \Gamma_n, t2T}$. The DR cost of a flexible load of this type, adapted from [32], is defined as

$$d_{\gamma,n}(\mathcal{Y}_n) = \mathrm{l}_\gamma^1 \sum_{t2[\mathrm{t}_\gamma^{\text{arr}},\mathrm{t}_\gamma^{\text{dep}}]} x_{\gamma,t} + \mathrm{l}_\gamma^2, \quad , \forall \gamma \in \Gamma_n, n \in N_{\text{flex}}. \tag{37}$$

5.1.5 Storage Facility

Set N_{bat} contains storage facilities. A battery is characterized by the charging and discharging efficiency parameters e_γ^c and e_γ^d, respectively, a maximum battery capacity $\overline{\mathrm{SOC}}_\gamma$, a maximum power rate $\overline{\mathrm{x}}_\gamma$ and a maximum number b_γ of full discharge cycles allowed. The set of control variables is $\mathcal{Y}_{n:n2N_{\text{bat}}} = \{x_{\gamma,t}^{ch}, x_{\gamma,t}^{dis}, u_{\gamma,t}, SOC_{\gamma,t}\}$, where for timeslot t, variable $x_{\gamma,t}^{ch}$ is the charge power, $x_{\gamma,t}^{dis}$ is the discharge power, $u_{\gamma,t}$ is a binary variable denoting whether γ charges or discharges, and $SOC_{\gamma,t}$ is the battery's state of charge. A storage facility does not have an operational cost for DR, i.e.,

$$d_{\gamma,n}(\mathcal{Y}_n) = 0, \quad \forall \gamma \in \Gamma_n, n \in N_{\text{bat}} \tag{38}$$

but the operation of a battery is subject to the following set of constraints [33]:

$$0 \leq x_{\gamma,t}^{ch} \leq u_{\gamma,t}\overline{\mathrm{x}}_\gamma \tag{39}$$

$$0 \leq x_{\gamma,t}^{dis} \leq (1-u_{\gamma,t})\overline{\mathrm{x}}_\gamma \tag{40}$$

$$SOC_{\gamma,t} = SOC_{\gamma,t-1} + \mathrm{e}_\gamma^c x_{\gamma,t}^{ch} - x_{\gamma,t}^{dis}/\mathrm{e}_\gamma^d \tag{41}$$

$$0 \leq SOC_{\gamma,t} \leq \overline{\mathrm{SOC}}_\gamma \tag{42}$$

$$SOC_{\gamma,jTj} \geq SOC_{\gamma,0} \tag{43}$$

$$\sum_{t2T} x_{\gamma,t}^{dis} \leq \mathrm{b}_\gamma \cdot \overline{\mathrm{SOC}}_\gamma. \tag{44}$$

5.1.6 Facility with Thermostatically Controlled Loads

Let N_{tcl} denote the set of facilities that feature thermostatically controlled loads (TCLs). Such facilities control the power consumption $x_{\gamma,t}$ of a TCL as well as indirectly controlling the room temperature $H_{\gamma,t}$, i.e. $\mathcal{Y}_{n:n2N_{\text{tcl}}} = \{x_{\gamma,t}, H_{\gamma,t}\}$. A TCL is characterized by minimum and a maximum acceptable temperature levels, denoted as $\underline{\mathrm{H}}_{\gamma,t}$ and $\overline{\mathrm{H}}_{\gamma,t}$ respectively. The TCL's temperature must be within $[\underline{\mathrm{H}}_{\gamma,t}, \overline{\mathrm{H}}_{\gamma,t}]$ at all times:

$$\underline{\mathrm{H}}_{\gamma,t} \leq H_{\gamma,t} \leq \overline{\mathrm{H}}_{\gamma,t}, \quad \forall t \in T, \gamma \in \Gamma_n, n \in N_{\text{tcl}}. \tag{45}$$

The temperature transition depends on technical parameters $\mathrm{h}_\gamma^{\text{ins}}, \mathrm{h}_\gamma^{\text{eff}}$ of the TCL that relate to the room's insulation and the TCL's efficiency, as well as on the TCL's initial temperature $\mathrm{H}_{\gamma,0}$ and the outdoors temperature $\mathrm{H}_t^{\text{out}}$, as in

$$H_{\gamma,t} = (1-\mathrm{h}_\gamma^1)^t \mathrm{H}_{\gamma,0} - \sum_{\tau=1}^{t} (1-\mathrm{h}_\gamma^{\text{ins}})^{(t-\tau)} \mathrm{H}_t^{\text{out}} + \sum_{\tau=1}^{t} (1-\mathrm{h}_\gamma^{\text{ins}})^{(t-\tau)} \mathrm{h}_\gamma^{\text{eff}} x_{\gamma,t} \tag{46}$$

The DR cost function of a TCL is defined as the distance from its desired setpoint temperature $\hat{\mathrm{H}}_{\gamma,t}$ [34], i.e.,

$$d_{\gamma,n}(\mathcal{Y}_n) = \mathrm{h}_\gamma^{\text{cost}} (\hat{\mathrm{H}}_{\gamma,t} - H_{\gamma,t})^2, \quad \forall n \in N_{\text{tcl}}. \tag{47}$$

5.1.7 Electric Vehicles Charging Station

An EV charging station $n \in N_{\text{cs}}$ can form a flexible facility by scheduling the power consumption of its charging tasks. An EV $\gamma \in \Gamma_{n:n2N_{\text{cs}}}$ is characterized by an arrival time $\mathrm{e}_\gamma^{\text{arr}}$, an energy requirement E_γ, a charging efficiency parameter $\mathrm{e}_\gamma^{\text{eff}}$ and a maximum charging rate $\overline{\mathrm{x}}_\gamma$. The set of variables is $\mathcal{Y}_{n:n2N_{\text{cs}}} = \{x_{\gamma,t}, SOE_{\gamma,t}, u_{\gamma,t}\}$ where $SOE_{\gamma,t}$ is the state of energy in the EV's battery and $u_{\gamma,t}$ is a binary variable denoting whether the EV's charging demand has been satisfied in timeslot t. The set of constraints describing the EV model is:

$$x_{\gamma,t} = 0, \quad t < \mathrm{e}_\gamma^{\text{arr}} \tag{48}$$

$$\mathrm{e}_\gamma^{\text{eff}} \sum_{t2T} x_{\gamma,t} = \mathrm{E}_\gamma \tag{49}$$

$$SOE_{\gamma,t} = SOE_{\gamma,t-1} \mathrm{e}_\gamma^{\text{eff}} x_{\gamma,t} \tag{50}$$

$$u_{\gamma,t} = \begin{cases} 1, & SOE_{\gamma,t} - \mathrm{E}_\gamma < 0 \\ 0, & SOE_{\gamma,t} - \mathrm{E}_\gamma \geq 0 \end{cases} \tag{51}$$

TABLE 1
Technical characteristics of facilities

Parameters	Values
Curtailable loads	
$\underline{x}$ (kW)	0
$\overline{x}$ (kW)	[1, 5]
$\tilde{x}$ (%$\overline{x}$)	[70, 100]
c (€/kW^2)	[0.25, 0.50]
Curtailable loads With Ramp Constraints	
$\underline{x}$ (kW)	0
$\overline{x}$ (kW)	[10, 15]
c (€/kW^2)	[0.03, 0.05]
r^{down} (kW)	[4, 6]
r^{up} (kW)	[4, 6]
Time-shiftable Loads	
$\underline{x}$ (kW)	0
$\overline{x}$ (kW)	[2, 12]
E (kWh)	[6, 36]
t^{arr} (h)	[3, 9] [[14, 20]
t^{dep} (h)	[5, 11] [[16, 22]
$\tilde{t}$ (h)	t^{arr} + E =$\overline{x}$
s (€ h)	[1.0, 1.1]
Fully Flexible Loads	
$\underline{x}$ (kW)	0
$\overline{x}$ (kW)	2
t^{arr} (h)	[1, 21]
t^{dep} (h)	[1, 21] + [2, 5]
$\underline{E}$ (kWh)	[4.8, 5.1]
$\overline{E}$ (kWh)	[5.5, 6.0]
l^1 (€/kW)	[0.25, 0.75]
l^2 (€)	[2, 3]
Storage Facilities	
$\underline{x}$ (kW)	0
$\overline{x}$ (kW)	2.5
e^c (%)	95
e^d (%)	95
$\overline{SOC}$ (kWh)	5
$SOC_{;0}$ (kWh)	2.5
b (#)	2
Thermostatically Controlled Loads	
$\underline{x}$ (kW)	0
$\overline{x}$ (kW)	5
$\underline{H}$ (° F)	74
$\overline{H}$ (° F)	83
$\til{H}$ (° F)	79
h^{ins} (%)	90
h^{eff} (° F/kW)	-3
h^{cost} (€/ ° F^2)	[0.15, 0.20]

Then, the DR cost is defined based on the extra waiting time that an EV suffers due to delayed charging (beyond its earliest possible task completion time $\lceil \mathrm{E}_\gamma / \mathrm{e}_\gamma^{\mathrm{eff}} \overline{\mathrm{x}}_\gamma \rceil$):

$$d_{\gamma,n}(\mathcal{Y}_n) = \sum_{t2T} u_{\gamma,t} \cdot t - \lceil \mathrm{E}_\gamma / \mathrm{e}_\gamma^{\mathrm{eff}} \overline{\mathrm{x}}_\gamma \rceil - \mathrm{e}_\gamma^{\mathrm{arr}}, \quad \forall \gamma \in \Gamma_n, n \in N_{\mathrm{cs}}. \quad (52)$$

This formulation was first proposed in [35].

5.2 DSO network

We consider a 15 node radial distribution network (Fig. 2). The data for branches and loads are presented in Table 2, adopted by [33]. The upper and lower bounds of the nodal voltage amplitude are set to 1.05 pu and 0.95 pu, respectively. We assume that 2 PV generators are installed at nodes 2 and 13 of the network, while 4 wind turbines are located at nodes 5, 8, 10 and 11. Their production curves are derived from [36]. The base power and voltage are 1 MVA and 11kV. The cost c_a^{curt} of shedding 1 MW of RES generation was set to 50. For the evaluation we considered a number of 100 assets for each facility.

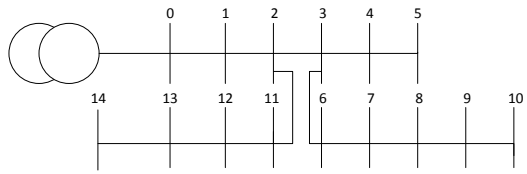

Fig. 2. A 15-node radial distribution network

5.3 Cloud computing infrastructure

In our simulation experiments, we considered two topologies for the COMNET infrastructure with different characteristics in terms of the number of available resources and link lengths: a basic (Fig. 3) and an extended (Fig. 4) topology. For both network topologies, we assumed that the network is split into three layers, with Layer 1 representing edge, Layer 2 fog, and Layer 3 cloud infrastructure. The link lengths of the basic topology vary from 100 km to 500 km, whereas the extended topology features average link lengths of 150 km that vary on the interval [30-500] km. The number, processing capacity and availability of the resources increase as we move to higher layers of the infrastructure (deeper in the cloud). We assume uniform processing capabilities at each node of a given layer. For the bottom layer, the processing capacity of a node was set to 9 GIPS. On the other hand, the utilization cost of the processing resources decreases from the edge to the cloud. The nodes of the different layers are interconnected through links of varying rates. Edge nodes are connected via lower rate links, while cloud nodes via higher speed links. However, in our performed simulation experiments the transmission latency was assumed to be negligible, given the small size of data that need to be transferred, and only the propagation latency was taken into consideration.

We examined the performance of the proposed ILP and heuristic assuming an instance of the DROaaS problem, in which a varying number of DR requests [10-60] need to be served. Each DR request refers to a certain set of facilities, the number of which was selected randomly from [2, 8]. We assumed that higher layer resources decrease the execution time of a task by 20% and the cost of utilizing processing power by 40% (1 c.u. for using layer 1 for 10 sec), in relation to lower layer resources (cloud-fog and fog-edge). The processing capacity of each node of the edge/fog/cloud

TABLE 2
Technical characteristics of the 15-node radial distribution network

Branch (#)	From Node (i)	To Node (a)	R_{ia} (pu)	X_{ia} (pu)	$\overline{P}_{ia}; \overline{Q}_{ia}$ (pu)	$\underline{P}_{ia}; \underline{Q}_{ia}$ (pu)	$\overline{V}_{ia}$ (pu)	$\underline{V}_{ia}$ (pu)
1	0	1	0.0031	0.0752	0.233	-0.233	0.95	1.05
2	1	2	0.0033	0.0018	0.233	-0.233	0.95	1.05
3	2	3	0.0067	0.0308	0.233	-0.233	0.95	1.05
4	3	4	0.0058	0.0149	0.233	-0.233	0.95	1.05
5	4	5	0.0141	0.0365	0.233	-0.233	0.95	1.05
6	6	6	0.0080	0.0369	0.233	-0.233	0.95	1.05
7	6	7	0.0090	0.0415	0.233	-0.233	0.95	1.05
8	7	8	0.0070	0.0323	0.233	-0.233	0.95	1.05
9	8	9	0.0037	0.0169	0.233	-0.233	0.95	1.05
10	9	10	0.0090	0.0415	0.233	-0.233	0.95	1.05
11	2	11	0.0275	0.1270	0.233	-0.233	0.95	1.05
12	11	12	0.0315	0.0814	0.233	-0.233	0.95	1.05
13	12	13	0.0396	0.1029	0.233	-0.233	0.95	1.05
14	13	14	0.0106	0.0041	0.233	-0.233	0.95	1.05

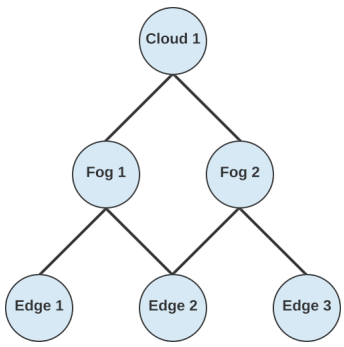

Fig. 3. Basic network topology, split into 3 Layers to represent an edge-fog-cloud infrastructure

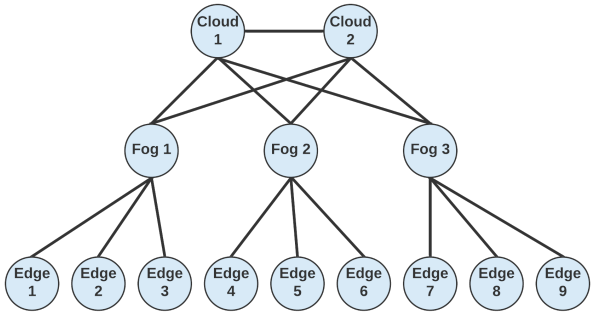

Fig. 4. Extended network topology, split into 3 Layers to represent an edge-fog-cloud infrastructure

layer was set to 9/10.8/12.96 GIPS respectively, while the respective cost of using resources for 10 sec was set to 1/1.4/1.96.

6 Results

Simulation experiments were performed, evaluating different scenarios in relation to the number of tasks, their processing and data requirements, the capacities of the computing and networking resources and their related costs. The computational load of line 4 of Algorithm 1, i.e. for solving the optimization problems (15) and (14) for each different facility type, were tested via simulations. The results are presented in Table 3. An interesting observation is that the computational cost for curtailable loads is massively increased by the sole introduction of ramp constraints. Also, the network loads, which relate to the volume of data necessary (number of parameters) to perform the calculations, are presented in Table 4 for each facility type. Finally, we should note that obtaining the optimal solution to problem (18)-(27) takes a prohibitive amount of time (in the order of hours). In contrast, the computational time of the Heuristic algorithm is only in the order of seconds. This makes the Heuristic algorithm applicable for the purposes of real-time electricity markets, which are typically cleared every 5 to 15 minutes. In what follows, we present simulation experiments that record the optimality loss of the fast Heuristic algorithm, compared to the optimal, but impractical, ILP.

TABLE 3
Number of instructions of line 4 of Algorithm 1, for the DSO and each facility type

Facility type	Instructions count (Millions)
DSO	16703
Curtailable loads	19369
Curtailable loads with ramps	874284
Time-shiftable loads	9028
Flexible loads	9026
Storage	15895
Thermostatically Controlled Loads	196891
EV charging station	120714

6.1 Results for the basic network topology

Initially, we evaluated the performance of the proposed ILP and heuristic mechanisms in relation to the total cost required to complete the execution of an iteration of the DR requests (Fig. 5). As expected, lower cost is achieved when the objective is set to minimize the processing per iteration cost ($w = 1$). In that case, the cloud resource nodes are preferred compared to the edge and fog nodes due to their lower cost and higher

TABLE 4
Network load (in number of parameters needed to be communicated) for the DSO and each facility type

Facility type	Input (4*bytes*)	Output (4*bytes*)
Curtailable loads	5000	2400
Curtailable loads with ramps	5400	2400
Time-shiftable loads	700	2400
Flexible loads	800	2400
Storage	600	4800
Thermostatically Controlled Loads	7924	2400
EV charging station	600	2400

processing capabilities. The performance of the proposed ILP and heuristic is similar for a small number of DR requests, while for a higher number of DR requests the ILP outperforms the heuristic. When the objective is the minimization of the latency for serving DR requests ($w = 0$), then more edge resources are utilized, resulting in increased cost when the heuristic mechanism is used. For these experiments, the latency bound for each DR request was set to 1.3 times the processing time of the largest task on the slowest resource.

Next, we compared the total time required to complete an iteration (makespan) (Fig. 6) for the two developed mechanisms. In this case, the latency bound is relaxed. The best performance is achieved by the ILP with the objective of minimizing the latency for serving the DR requests, followed by the respective heuristic. The difference between the heuristic and the ILP is due to the fact that the ILP achieves the optimal allocation of the processing resources, while the heuristic with a worse performance in resource utilization, selects processing instances with slower computational capabilities to meet the objective criteria. All simulation experiments were performed using Matlab and the CPLEX LP/MIP solver.

6.2 Results for the extended network topology

We performed a number of experiments for the extended topology of Fig. 4, using the heuristic algorithm for two different cases. In the first case, we assumed a higher number of DR requests that vary from 200 to 1200, while the rest of our assumptions remained the same as in the basic network topology. In this case, we examined the allocation of resources at the different layers under the two objectives.

As shown in Fig. 7, when the objective is the minimization of the processing cost (left bar), more tasks are executed in Layer 3. Also, the number of tasks executed in Layer 3 increases as the total number of DR request increase, taking advantage of the higher number and more powerful computational resources that are available in the cloud. In this case, the utilization of the Layer 2 resources is low. On the other hand, when the objective is the minimization of the service completion time (right bar), Layer 1 and Layer 2 resources are preferred. Especially the latter ones are highly utilized because of their advantage in accommodating the tasks

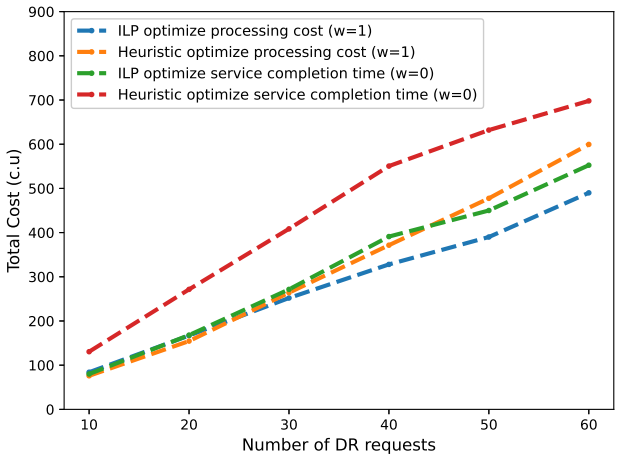

Fig. 5. Total cost required to complete the execution of an iteration of the DR requests for the basic network topology

that cannot be served by the Layer 1 resources due to the high waiting time that would violate the latency constraint. Hence, when the main optimization criterion is the cost, the cloud resources are the most appropriate ones, but when the objective is the minimization of the service time, edge and fog resources are preferred as they offer shorter delays at the expense of a higher cost.

In the second case examined, we assumed the use of special purpose hardware accelerators, such as GPUs, in the edge layer that provide a performance boost of 30% for the execution of DR facility tasks, compared to the general-purpose resources present in the fog and cloud layers. When the more powerful equipment is present at the edge, the edge resources are preferred under both objectives and tend to achieve significantly better performance compared to the case with our initial assumptions (Fig. 8). This is because the enhanced edge devices complete the tasks faster, as is depicted in both the processing cost and the total time required to complete an iteration. On the other hand, when no enhanced equipment is used at the edge, the completion time lags behind by 23% and 27% for the completion time and processing cost objectives, respectively.

7 Conclusion

In this paper, we considered the problem of clearing a DR flexibility market of a power distribution network using a diverse set of computational resources (edge, fog, cloud) over the cloud continuum. We presented a flexibility market clearing algorithm based on Lagrangian relaxation, and we configured the algorithm's execution with an optimal computational resource allocation algorithm. The resource allocation algorithm is able to service multiple demand response flexibility markets, and leverage an economy-of-scale effect towards minimizing the cost of computational resources while respecting the execution time constraints of each request. Our experimental results demonstrate the effect of different DR

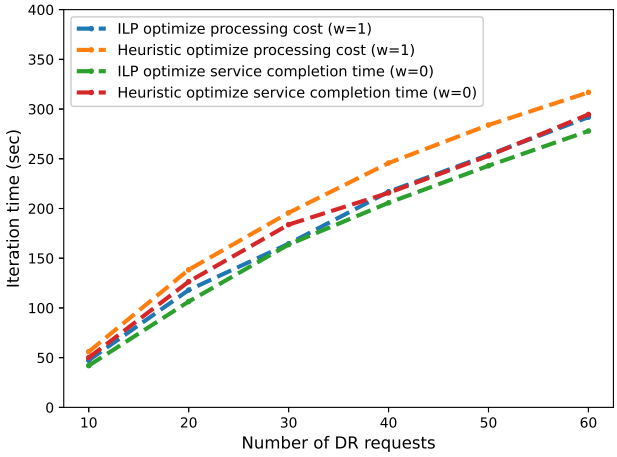

Fig. 6. Total time required to complete an iteration for the ILP and the heuristic mechanism

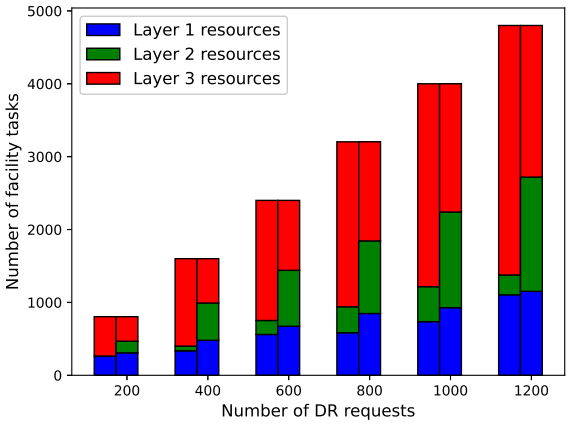

Fig. 7. Resource allocation at the different network layers for the two objectives ($\mathrm{w} = 1$ for the left bar, and $\mathrm{w} = 0$ for the right bar) and the heuristic algorithm

models in the resulting computational burden (e.g, the sole introduction of ramp constraints had a dramatic effect), the trade-off between optimality and scalability (as approached by the optimal solution and a faster but sub-optimal heuristic), as well as the resulting allocation of computational tasks through the different layers (edge / fog / cloud) of the envisaged architecture.

Our work enables a new business model, where a service provider can offer the functionality of DR operation as a service. An important direction for future work, would be to compare different decomposition methods with respect to their performance/interaction with various types of computational and networking infrastructure. A more general milestone refers to the development of an integrated and holistic DR system able to combine orthogonal technologies in order to offer an attractive trade-off between cost-efficiency and scalability, while also preserving the participants' privacy.

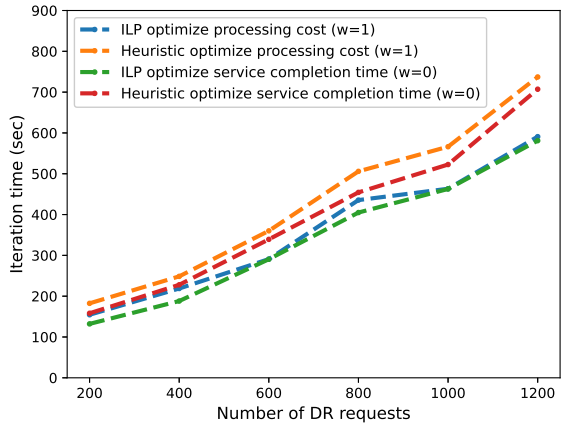

Fig. 8. Total time required to complete an iteration for the ILP and the heuristic for the extended network with enhanced edge resources

References

[1] K. Seklos, G. Tsaousoglou, K. Steriotis, N. Efthymiopoulos, P. Makris, and E. Varvarigos, "Designing a distribution level flexibility market using mechanism design and optimal power flow," in *2020 International Conference on Smart Energy Systems and Technologies (SEST)*, 2020, pp. 1–6.

[2] USEF, "A flexibility market design," https://www.usef.energy/download-the-framework/a-flexibility-market-design/.

[3] "Flexgrid project h2020," https://flexgrid-project.eu/, 2020.

[4] X. Jin, Q. Wu, and H. Jia, "Local flexibility markets: Literature review on concepts, models and clearing methods," *Applied Energy*, vol. 261, p. 114387, 2020.

[5] G. Tsaousoglou, J. S. Giraldo, P. Pinson, and N. G. Paterakis, "Mechanism design for fair and efficient dso flexibility markets," *IEEE Transactions on Smart Grid*, pp. 1–1, 2020.

[6] K. Steriotis, G. Tsaousoglou, N. Efthymiopoulos, P. Makris, and E. M. Varvarigos, "A novel behavioral real time pricing scheme for the active energy consumers' participation in emerging flexibility markets," *Sustainable Energy, Grids and Networks*, vol. 16, pp. 14–27, 2018.

[7] S. Bera, S. Misra, and J. J. Rodrigues, "Cloud computing applications for smart grid: A survey," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 5, pp. 1477–1494, 2014.

[8] M. Yigit, V. C. Gungor, and S. Baktir, "Cloud computing for smart grid applications," *Computer Networks*, vol. 70, pp. 312–329, 2014.

[9] X. Fang, D. Yang, and G. Xue, "Evolving smart grid information management cloudward: A cloud optimization perspective," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 111–119, 2013.

[10] E. Dall'Anese, K. Baker, and T. Summers, "Chance-constrained ac optimal power flow for distribution systems with renewables," *IEEE Transactions on Power Systems*, vol. 32, no. 5, pp. 3427–3438, 2017.

[11] T. Soares, R. J. Bessa, P. Pinson, and H. Morais, "Active distribution grid management based on robust ac optimal power flow," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6229–6241, 2017.

[12] M.-M. Zhao, Q. Shi, Y. Cai, M.-J. Zhao, and Y. Li, "Distributed penalty dual decomposition algorithm for optimal power flow in radial networks," *IEEE Transactions on Power Systems*, vol. 35, no. 3, pp. 2176–2189, 2019.

[13] X. Pan, A. Jiang, and H. Wang, "Edge-cloud computing application, architecture, and challenges in ubiquitous power internet of things demand response," *Journal of Renewable and Sustainable Energy*, vol. 12, no. 6, p. 062702, 2020.

[14] C. Feng, Y. Wang, Q. Chen, G. Strbac, and C. Kang, "Smart grid encounters edge computing: Opportunities and applications." *Advances in Applied Energy*, p. 100006, 2020.