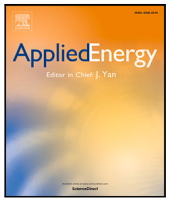
Applied Energy 343 (2023) 121234

Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy



Operating peer-to-peer electricity markets under uncertainty via learning-based, distributed optimal control

Georgios Tsaousoglou [a,*], Petros Ellinas [b], Emmanouel Varvarigos [b]

[a] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark
[b] National Technical University of Athens, Institute of Communications and Computer Systems, Greece

ARTICLE INFO

Keywords:
Distributed energy resources
Peer-to-peer electricity markets
Optimal control
Distributed stochastic optimization
Learn to optimize
ADMM

ABSTRACT

Towards the global endeavor of clean energy transition, there is a rapid development of distributed energy resources installed in the premises of residential or commercial users, enabling them to act as flexible energy prosumers. Empowering prosumers is envisioned as a catalytic development for modern energy economies, with recent research, as well as innovation and policy actions, pointing to the promising direction of decentralized energy markets, where active energy prosumers exchange energy in a decentralized fashion. Despite the vast amount of recent research on prosumer-centric peer-to-peer (p2p) energy markets, only a small subset of studies accounts for managing the inherent uncertainty of prosumers' flexible demands.

In this paper, we consider the problem of controlling the decisions of energy prosumers' within a p2p exchange network. The multi-bilateral economic dispatch is formulated as an optimal control problem. The proposed solution is based on a direct lookahead policy, effectively addressing the issues of dimensionality and local constraint satisfaction. Experimental simulations demonstrate the method's efficiency and the system's behavior. The proposed formulation and method is shown to effectively address the operation of p2p markets under uncertainty, closely tracking the performance of the (full information) optimal-in-hindsight benchmark.

1. Introduction

The transition to clean energy is triggering innovative developments in various aspects of the power and energy systems.

Decentralized electricity markets is a major such development, through which the focus is shifted from centralized bulk power plants to active energy prosumers generating and consuming energy, as well as exchanging energy among each other. Such decentralized market architectures empower prosumers, by engaging them in an active role in power systems while letting them discover (and be attributed) the value of their energy [1]. This, in turn, acts as an incentive towards further micro-investments in distributed energy resources, accelerating the clean energy transition.

From a technical perspective, these decentralized markets call for distributed algorithmic techniques towards performing the necessary market clearing actions in a fast and efficient manner. More specifically, a set of prosumers along with their multiple bilateral connections for energy exchange, can be visualized as a graph, while each prosumer features certain energy generating/consuming facilities each one bearing a cost (or utility) function. The peer-to-peer (p2p) market clearing problem refers to deciding how much energy will be generated and consumed by each prosumer, as well as the amount of energy exchanges through each bilateral link. The efficiency of the market clearing process refers to maximizing the social welfare (or, equivalently, minimizing the aggregate cost) of the system. For elaborate presentations of p2p electricity markets the reader can refer to comprehensive recent surveys [2,3]. In this paper, we contribute to the related literature by formulating the p2p market clearing as an optimal control problem, accounting also for the prosumers' inherent uncertainties.

1.1. Related work

The problem of p2p market clearing was comprehensively modeled and analyzed by authors in [4,5], using a distributed optimization framework. In this framework, the market clearing problem takes the form of a multi-bilateral economic dispatch problem, which can be solved in a distributed manner by relaxing the coupling constraints between pairs of prosumers using Lagrangian relaxation. The optimal Lagrange multipliers of these constraints can be reached through iterative updates, namely using the Alternating Direction Method of

* Corresponding author.
E-mail address: geots@dtu.dk (G. Tsaousoglou).

https://doi.org/10.1016/j.apenergy.2023.121234
Received 18 August 2022; Received in revised form 13 April 2023; Accepted 29 April 2023
Available online 12 May 2023

Multipliers (ADMM), and be interpreted as prices for the bilateral energy exchange. An improved version of the ADMM algorithm for prosumer-based systems was proposed in [6], where the authors train a neural network to learn the system's Lagrange multipliers. Moreover, such exchanges can be configured with distributed ledger technologies towards integrating the decentralized conceptualization of such frameworks [7–9]. Notably, significant policy developments have been undertaken towards this direction [10], triggering a rapid growth in projects involving p2p energy markets [11].

State-of-the-art literature examines how such p2p markets can be configured with the safe operation of the physical underlying distribution network. Namely, [12] focuses on minimizing the power losses created by p2p exchanges and allocating their financial value, while [13] incorporates the power transfer limits of physical distribution network lines in the decentralized optimization problem. Another impactful direction refers to generalizing the notion of prosumers into small energy hubs that feature resources of various energy carriers (namely gas, heat and possibly hydrogen in addition to only electricity). Ref. [14] models such multi-energy prosumers and analyzes the fairness properties of the p2p market dispatch, while the latest work in [15] presented a scalable distributed optimization algorithm for a p2p market with prosumers spanning electricity and heat carriers. Moreover in [16], a distributed energy trading mechanism has been proposed, that allows energy exchange between multiple types of grid participants and energy hubs.

Another recent research direction considers the prosumers' strategic market participation and formulates the multi-bilateral interaction using game-theoretic models and concepts. The work in [17] modeled the problem as a mean-field game considering given (fixed) energy demands/offers from prosumers and a simple matching mechanism. The authors in [18] present a comprehensive analysis of a p2p market (generalized Nash) equilibrium, including a characterization of the price of anarchy. Bilateral exchanges are considered jointly with exchanges with the main grid, using a game-theoretic approach, in a hybrid market framework in [19]. Finally, putting the focus back on physical network constraints, the work in [20] studies the generalized Nash equilibrium in the presence of power flow constraints.

Nevertheless, the prosumers' energy consumption needs during the day are subject to uncertainty, a point not considered in formulations based on the deterministic models discussed so far. Making energy management decisions under uncertainty, especially in the case of flexible prosumers, has received great attention in the smart grid literature (see [21] for a recent review). However, uncertainty management within a p2p market framework is yet to be addressed, with only a few recent works considering some form of uncertainty within p2p markets. The authors in [22,23], consider uncertainty (within stochastic bi-level programs) using scenarios for uncertainty realization. This technique is shown to be effective in p2p exchanges between wind producers and *aggregators* of prosumers. Nevertheless, in fully decentralized markets with multiple individual prosumers (and therefore multiple sources of uncertainty) the scenario tree grows very abruptly, constituting the method unsuitable for online prosumer energy management and p2p market operation. Towards enabling fast online decision making, the authors in [24] propose an online optimization approach for p2p market operation which, however, does not fully exploit the statistical information over the prosumers' uncertain parameters that is usually available. Ref. [25] treats uncertainty by repeatedly solving a deterministic, p2p market, optimization problem, in a model predictive control fashion. However, this approach is after adapting to observed (realized) uncertainties, rather than proactively accounting for them. Finally, the work in [26] configures the model predictive control approach with forecasts for uncertain parameters, adopting the "forecast-first, then-optimize" approach for the p2p market operation.

1.2. Research gap, paper contributions and organization

The main shortcoming of most works in the p2p electricity market literature is the complete absence of uncertainty consideration, while the few papers that consider uncertain factors adopt a simplified model (e.g. a deterministic-equivalent optimization) as described towards the end of Section 1.1. More generally, it appears that no prior study on p2p electricity markets has addressed the need for an uncertainty-aware, sequential dispatch process, as revealed by the summary of the literature review presented in Table 1.

In this paper, we propose a novel technique towards operating a p2p prosumer-based market under uncertainty of flexible prosumer demands. Assuming that statistical information over such uncertainties is available, we consider an end-to-end approach where, instead of learning to forecast the uncertain parameters (and then optimize the system), the learning module is after learning to directly optimize the system (in our context, learning an optimal policy towards deciding the prosumers' generation, consumption and exchanges). A closely-related concept is known as the learn-to-optimize approach in some communities (e.g. [27]).

The standard learn-to-optimize approach parameterizes and trains a machine-learning (ML) module, to predict an optimal decision for the decision variables. In the language/framework set by [28], this is effectively a policy-function approximation. In our multi-bilateral prosumer market, however, we show that this approach entails a very large amount of decision variables, rendering the learning task inefficient. Moreover, in the approach where the ML directly controls each prosumer's energy management decisions, the satisfaction of individual prosumer constraints is not guaranteed.

Our proposed solution, combining the deterministic Lagrangian optimization methods discussed with the powerful ML machinery, is to perform the learning on the (much smaller) dual problem's space. More specifically, we learn to predict the system's optimal dual variables using a ML algorithm (trained offline), once the system's state is observed. These future dual variables (i.e. the Lagrange multipliers of the bilateral exchanges) are shared with the prosumers. Finally, each prosumer's control actions are decided in a distributed manner using the ADMM algorithm, but with the twist that the Lagrange multipliers of future times are set as predicted by the ML algorithm. In the proposed approach, each prosumer controls its own resources, making sure that its local constraints are respected, while its decisions are informed of the available information (prediction) about the future. Using a predicted future system trajectory classifies the method as a direct look-ahead policy,[1] with the special feature that all information about the future is compactly encapsulated into a set of multipliers allowing each prosumer to distributedly decide its own controls.

The paper's contributions can be summarized as follows:

- We formulate a p2p economic dispatch problem as an optimal control problem, taking into account the system's uncertainty.
- In contrast to stochastic programming techniques that cannot handle problems where the uncertainty dimension increases exponentially with the number of decision stages, we consider a machine learning algorithm towards a learn-to-optimize approach.
- In contrast to standard learn-to-optimize approaches, we propose learning the system's dual variables instead of the primal and use them as predicted prices for future p2p exchanges. This allows the prosumers the freedom to decide on their actions, while making sure that their local constraints will be respected.

[1] The family of direct look-ahead policies refers to techniques that approximate the possible future system trajectories using a (typically deterministic) model based on current information. In this way, the control actions are decided by solving a deterministic optimization problem formulated to incorporate the look-ahead model [28]. Model predictive control would be a predominant example of a direct look-ahead policy.

Table 1
Literature classification based on the motivated attributes of the Peer to Peer decision-making process.

	Uncertainty consideration	Sequential decisions	Data-Driven	Distributed computations
[4], [5], [12],[13], [16] [17], [18], [19], [20]	×	×	×	✔
[6]	✔	×	✔	✔
[22],[23],[24], [25],[26]	✔	×	×	×
this work	✔	✔	✔	✔

The system model and problem formulation are presented in Section 2. In Section 3, we unfold the presentation of the proposed control policy. Section 4 presents the simulation setup and the experimental evaluation of our proposition, while Section 5 concludes the paper.

2. System model and problem formulation

Let us consider a set $\mathcal{N}$ of prosumers, partly inteconnected via a communication graph where a prosumer $n \in \mathcal{N}$ interacts with its connected prosumers (peers) of the set $\mathcal{C}_n \subset \mathcal{N}$. We are interested in the energy management and exchange decisions of set $\mathcal{N}$ within a set $\mathcal{T}$ of discrete timeslots.

2.1. System model

At a timeslot $t \in \mathcal{T}$, a prosumer $n \in \mathcal{N}$ can generate an amount $p^{\mathrm{g}}_{n,t}$ of energy, bearing a generation cost $c^{\mathrm{g}}_{n,t}$ modeled as

$$c^{\mathrm{g}}_{n,t} = \mathrm{a}_n (p^{\mathrm{g}}_{n,t})^2 + \mathrm{b}_n p^{\mathrm{g}}_{n,t} + \mathrm{d}_n, \quad \forall n \in \mathcal{N}, t \in \mathcal{T}, \tag{1}$$

where $\mathrm{a}_n, \mathrm{b}_n, \mathrm{d}_n$ are positive constants. This quadratic model covers both the case in which the prosumer features generation facilities (where the convex cost function models the decreasing fuel efficiency of the generator), as well as the case of prosumer load curtailments, in which the quadratic term implements a convex relationship between cost and amount of curtailment; notice that this is equivalent to a concave relationship between energy consumption and user utility, as prescribed by microeconomic theory.

The prosumer also features a set $\mathcal{D}_n$ of flexible demands. A flexible demand $d \in \mathcal{D}_n$ is characterized by a tuple $\Omega_d = (\mathrm{t}^{\mathrm{a}}_d, \widetilde{\mathrm{t}}^{\mathrm{d}}_d, \mathrm{t}^{\mathrm{d}}_d, \mathrm{E}_d, \overline{\mathrm{p}}_d, \mathrm{w}_d)$, comprised by its arrival time $\mathrm{t}^{\mathrm{a}}_d$, its desired departure time $\widetilde{\mathrm{t}}^{\mathrm{d}}_d$, its deadline $\mathrm{t}^{\mathrm{d}}_d > \widetilde{\mathrm{t}}^{\mathrm{d}}_d$, its energy requirement E_d, its maximum consumption rate $\overline{\mathrm{p}}_d$, and its elasticity w_d that captures how impatient the demand is. The energy allocated to flexible demand $d \in \mathcal{D}_n$ at t is denoted as $p^{\mathrm{f}}_{d,t}$, and the allocation profile needs to satisfy the demand within $[\mathrm{t}^{\mathrm{a}}_d, \mathrm{t}^{\mathrm{d}}_d]$, as in

$$\sum_{t \in [\mathrm{t}^{\mathrm{a}}_d, \mathrm{t}^{\mathrm{d}}_d]} p^{\mathrm{f}}_{d,t} = \mathrm{E}_d, \quad \forall d \in \mathcal{D}_n, n \in \mathcal{N}, \tag{2}$$

while a demand can only consume an amount of energy between zero and its maximum consumption rate, i.e.

$$0 \le p^{\mathrm{f}}_{d,t} \le \overline{\mathrm{p}}_d, \quad \forall d \in \mathcal{D}_n, n \in \mathcal{N}, t \in \mathcal{T}. \tag{3}$$

Importantly, we assume that the upper bounds $\overline{\mathrm{p}}_d$ are the result of a network feasibility study conducted by the system operator. This means that all possible demands below $\overline{\mathrm{p}}_d$ are pre-approved to satisfy the physical network's constraints and no real-time power flow model is needed. Note that this is the current practice in real systems.

Observe that the prosumer's inflexible demand can readily be modeled as a flexible demand with $\mathrm{E}_d = \overline{\mathrm{p}}_d \cdot (\mathrm{t}^{\mathrm{d}}_d - \mathrm{t}^{\mathrm{a}}_d)$. Moreover, when part of the demand is satisfied beyond the demand's desired departure time $\widetilde{\mathrm{t}}^{\mathrm{d}}_d$, there is a dissatisfaction cost $c^{\mathrm{d}}_{d,t}$ modeled as

$$c^{\mathrm{d}}_{d,t} = \begin{cases} \frac{p^{\mathrm{f}}_{d,t} \cdot \mathrm{w}_d^{\,t - \widetilde{\mathrm{t}}^{\mathrm{d}}_d}}{\mathrm{E}_d}, & \forall d \in \mathcal{D}_n, n \in \mathcal{N}, t > \widetilde{\mathrm{t}}^{\mathrm{d}}_d, \\ 0, & \forall d \in \mathcal{D}_n, n \in \mathcal{N}, t \le \widetilde{\mathrm{t}}^{\mathrm{d}}_d. \end{cases} \tag{4}$$

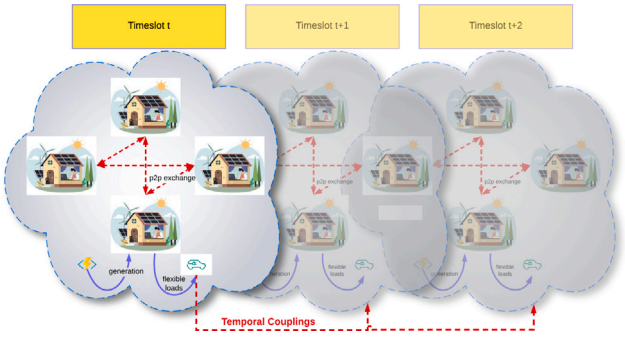


Fig. 1. Graphical illustration of the system's architecture.

A prosumer can self-consume energy (i.e. satisfy its demands from its own generation), and/or also exchange energy with its neighbors $m \in \mathcal{C}_n$. Let $p_{nm,t}$ denote the energy that prosumer n sends to its peer $m \in \mathcal{C}_n$ at timeslot $t \in \mathcal{T}$ (negative for receiving energy from m). Possible upper/lower bounds on each exchange (e.g. due to regulation) are taken into account, as in

$$\underline{\mathrm{p}}_{nm} \le |p_{nm,t}| \le \overline{\mathrm{p}}_{nm}, \quad \forall n \in \mathcal{N}, m \in \mathcal{C}_n, t \in \mathcal{T}. \tag{5}$$

Based on the so far definitions, the intra-prosumer power balance equation, for each prosumer, reads as

$$p^{\mathrm{g}}_{n,t} = \sum_{d \in \mathcal{D}_n} p^{\mathrm{f}}_{d,t} + \sum_{m \in \mathcal{C}_n} p_{nm,t}, \quad \forall n \in \mathcal{N}, t \in \mathcal{T}. \tag{6}$$

Finally, the inter-prosumer power balance constraints demand that the energy sent from n to m equals the energy that m received from n, i.e.

$$p_{nm,t} = -p_{mn,t}, \quad \forall n, m \in \mathcal{N}, t \in \mathcal{T}. \tag{7}$$

We consider prosumers with preferences over which peer they exchange energy with. Namely, a prosumer n bears a per-unit cost q_{mn} for buying energy from $m \in \mathcal{C}_n$. Thus, the exchange cost $c^{\mathrm{ex}}_{n,t}$ of n is given by

$$c^{\mathrm{ex}}_{n,t} = -\sum_{m \in \mathcal{C}_n} \mathrm{q}_{mn} p_{nm,t}, \quad \forall n \in \mathcal{N}, t \in \mathcal{T}. \tag{8}$$

2.2. Problem formulation

At any given timeslot t, the decisions on generation, consumption, and p2p exchanges, for all prosumers, need to be made by also taking into account the future, since the flexible loads constitute the system temporally coupled. The system's architecture is graphically illustrated in Fig. 1.

Our goal is to establish a control policy $\boldsymbol{\pi} = (\pi)_{n \in \mathcal{N}}$ for variables $\boldsymbol{u}_t \triangleq \left(p^{\mathrm{g}}_{n,t}, (p^{\mathrm{f}}_{d,t})_{d \in \mathcal{D}_n}, (p_{nm,t})_{m \in \mathcal{C}_n}\right)_{n \in \mathcal{N}}$, i.e. for all the decision variables $p^{\mathrm{g}}_{n,t}, p^{\mathrm{f}}_{d,t}, p_{nm,t}$ of each prosumer at any given timeslot. The system's *state* captures all the information that is relevant upon deciding a control action, and is defined as

$$\boldsymbol{x}_t \triangleq \left(t, \left((\Omega_d)_{d \in \mathcal{D}_n}\right)_{n \in \mathcal{N}}, \left(\sum_{\tau \in [\mathrm{t}^{\mathrm{a}}_d, t-1]} p^{\mathrm{f}}_{d,\tau}\right)_{d \in \mathcal{D}_n, n \in \mathcal{N}}\right), \tag{9}$$

where the first state variable is the current operational timeslot, the second set of variables are the tuples Ω_d of all demands of all prosumers, and the third is the up-to-now energy allocated to each demand.

For demands that are inactive at t (have departed or have not yet arrived), we set the values of Ω_d to a default null state. The arrival of new demands at t constitutes the system's *disturbance* $\boldsymbol{w}$. Finally, the system's total cost, when transitioning out of timeslot t, is defined as

$$c_t = \sum_{n\in\mathcal{N}} \left(c^{\mathrm{g}}_{n,t} + \sum_{d\in\mathcal{D}_n} c^{\mathrm{d}}_{d,t} + c^{\mathrm{ex}}_{n,t} \right) \tag{10}$$

Our objective can then be formulated as an optimal control problem [29], of the form:

$$\min_{\boldsymbol{\pi}} \left\{ \mathbb{E}\Big[\sum_{t\in\mathcal{T}} c_t \Big] \right\} \tag{11}$$

$$\text{s.t.} \quad \boldsymbol{x}_{t+1} = f(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{w}_t).$$

(1)–(10).

In the next Section, we present our approach for discovering an efficient policy in this multi-agent optimal control problem.

3. Solution approach

In this Section, we present the proposed approach towards discovering a well-performing policy for problem (11). To this end, we exploit a special property of the problem: while in generic optimal control problems the disturbances $\boldsymbol{w}_t$ are a (probabilistic) function of the state and control, as in $\boldsymbol{w}_t \in \mathcal{P}(\cdot|\boldsymbol{x}_t, \boldsymbol{u}_t)$, the system at hand features the special property of *exogenous*, or control-independent, disturbances $\boldsymbol{w}_t$, since new demand arrivals (flexible or inflexible alike) do not depend on the power allocation decisions but follow an independent stochastic process. This allows us to generate samples of optimal controls for given disturbance trajectories, relatively easily: first, we generate a set $\mathcal{S}$ of scenarios for the disturbance realization, with each scenario $s \in \mathcal{S}$ representing a random walk in the space of possible disturbance trajectories, i.e.

$$s \triangleq [\boldsymbol{w}_t]_{t\in\mathcal{T}}, \tag{12}$$

and then we obtain the optimal control actions for each scenario by solving a deterministic optimization problem. These samples of optimal-in-hindsight decisions, will facilitate the construction of our policy.

In the following subsections, we present different policies of increasing sophistication, each one building on the concepts of the previous, thus concluding with the proposed policy in Section 3.3.

3.1. Point-forecast distributed optimization

Using a certain uncertainty realization scenario $s \in \mathcal{S}$ as a point-forecast, we can formulate a *deterministic* multi-bilateral economic dispatch optimization problem, defined as the minimization of the aggregated prosumers' costs under the exchange and demand-satisfying constraints described:

$$\min_{p^{\mathrm{g}}_{n,t}, p^{\mathrm{f}}_{d,t}, p_{nm,t}} \left\{ \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \left(c^{\mathrm{g}}_{n,t} + \sum_{d\in\mathcal{D}_n} c^{\mathrm{d}}_{d,t} + c^{\mathrm{ex}}_{n,t} \right) \right\}$$

$$\text{s.t. (1)–(8).} \tag{13}$$

Observe that problem (13) is a convex optimization problem that can be tackled efficiently. Moreover, the optimal solution can also be reached via a distributed algorithm, namely ADMM. Towards developing the presentation of the proposed policy for our original problem (11), it is useful to present the ADMM procedure for the deterministic problem (13), because it will be used as a building block for our full policy. Let us relax constraints (7), and consider the augmented Lagrangian of the deterministic problem (13) as:

$$\mathcal{L} = \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \Bigg(c^{\mathrm{g}}_{n,t} + \sum_{d\in\mathcal{D}_n} c^{\mathrm{d}}_{d,t} + c^{\mathrm{ex}}_{n,t} - \sum_{m\in\mathcal{C}_n} \left(\lambda_{nm,t}(p_{nm,t} + p_{mn,t}) - \frac{\rho}{2}(p_{nm,t} + p_{mn,t})^2 \right) \Bigg). \tag{14}$$

where $\lambda_{nm,t}$ is the Lagrange multiplier corresponding to constraint (7). In ADMM, each prosumer iteratively solves a local optimization problem (i.e. deciding only for its local variables $p^{\mathrm{g}}_{n,t}, p^{\mathrm{f}}_{d,t}, p_{nm,t}$. Let $p^{\mathrm{g}}_{n,t}[k], p^{\mathrm{f}}_{d,t}[k], p_{nm,t}[k]$ denote $n's$ decision at iteration k and $(\lambda_{nm,t}[k])_{n,m\in\mathcal{N},t\in\mathcal{T}}$ denote the tuple of all Lagrange multipliers at k. The local optimization problem of n at k, reads as

$$p^{\mathrm{g}}_{n,t}[k], p^{\mathrm{f}}_{d,t}[k], p_{nm,t}[k] = \operatorname{argmin}\Big\{ \mathcal{L} \Big\} \tag{15}$$

$$\text{s.t. (1)–(6), (8),}$$

$$p^{\mathrm{g}}_{i,t}, p^{\mathrm{f}}_{i,t}, p_{ij,t} = p^{\mathrm{g}}_{i,t}[k-1], p^{\mathrm{f}}_{i,t}[k-1], p_{ij,t}[k-1], \quad \forall i \neq n,$$

where the last constraint clarifies that for n's local problem, all the variables decided by other prosumers $i \neq n$ are fixed to the values determined by them in the previous iteration.[2] Each prosumer solves its local problem (15) in parallel and then the multipliers $\lambda_{nm,t}[k+1]$ are updated as

$$\lambda_{nm,t}[k+1] = \lambda_{nm,t}[k] + \rho \cdot (p_{nm,t}[k] + p_{mn,t}[k]), \quad \forall n, m \in \mathcal{N}, t \in \mathcal{T}, \tag{16}$$

$$\lambda_{mn,t}[k+1] = \lambda_{mn,t}[k] - \rho \cdot (p_{nm,t}[k] + p_{mn,t}[k]), \quad \forall n, m \in \mathcal{N}, t \in \mathcal{T}, \tag{17}$$

The procedure iterates until

$$\max_{n,m\in\mathcal{N},t\in\mathcal{T}} \{ p_{nm,t} + p_{mn,t} \} \leq \varepsilon, \tag{18}$$

i.e., the highest violation of constraint (7) is below a small upper bound.

3.2. Point-forecast distributed model predictive control

Determining the values of our control variables for the whole horizon using a point-forecast just before the first timeslot, as described in the previous subsection, is obviously an open-loop policy. Building our way towards the proposed policy, in this subsection we describe a model predictive control (MPC) extension of the distributed (still point-forecast) ADMM solution. In the distributed MPC policy, the ADMM algorithm is again executed, similarly to the previous subsection, but after the system transitions to the next timeslot (and observing the realized disturbances) the point-forecasts are updated and the ADMM procedure is run again to determine the new control actions (which will in general be different than the actions calculated before the disturbances were realized).

Let τ denote the current timeslot of online operation and $\widetilde{p}^{\mathrm{g}}_{n,\tau}, \widetilde{p}^{\mathrm{f}}_{n,\tau}, \widetilde{p}_{nm,\tau}$ denote the final decisions for τ, i.e. after the ADMM has converged. We can write n's local problem at any operational timeslot τ, by reusing our formulation of problem (15) and adding the constraint that past decisions cannot be changed[3]:

$$\left(p^{\mathrm{g}}_{n,t}[k], p^{\mathrm{f}}_{d,t}[k], p_{nm,t}[k]\right)_{t\in\mathcal{T}} = \operatorname{argmin}\Big\{ \mathcal{L} \Big\} \tag{19}$$

[2] Observe that problem (15) is indeed local in the sense that, with fixed variables $p^{\mathrm{g}}_{i,t}, p^{\mathrm{f}}_{i,t}, p_{ij,t}$, the costs (and constraints) of other prosumers $i \neq n$ become constants (and trivially true respectively). Thus they do not take part in n's local problem.

[3] This formulation is redundant, since it still uses variables of past timeslots only to fix them. However, it is very handy in terms of implementation, since it maintains the same implementation as problem (15), only adding a simple constraint.

Algorithm 1 Distributed point-forecast model predictive control

Initialize $\tau = 0$
While $\tau \in \mathcal{T}$
  Observe system state $\boldsymbol{x}_\tau$
  Generate a point-forecast for future disturbances
  Initialize $k = 1$, $(\lambda_{nm,\tau}[0])_{n,m\in\mathcal{N}} = 0$
  Initialize $(p^{\mathrm{g}}_{n,t}[0], p^{\mathrm{f}}_{d,t}[0], p_{nm,t}[0])_{t\in\mathcal{T}}$ at random
  while $\max_{n\in\mathcal{N},m\in\mathcal{C}_n,t\in\mathcal{T}}\{p_{nm,t}[k] + p_{mn,t}[k]\} > \varepsilon$:
    Each prosumer solves problem (19) to decide $p^{\mathrm{g}}_{n,t}[k], p^{\mathrm{f}}_{d,t}[k], p_{nm,t}[k]$
    Set $\lambda_{nm,t}[k+1]$ using (16)
    Set $k = k + 1$
  Set $(\widetilde{p}^{\mathrm{g}}_{n,\tau}, \widetilde{p}^{\mathrm{f}}_{n,\tau}, \widetilde{p}_{nm,\tau})_{n,m\in\mathcal{N}}$ to the converged values $(p^{\mathrm{g}}_{n,\tau}[k-1], p^{\mathrm{f}}_{d,\tau}[k-1], p_{nm,\tau}[k-1])_{n,m\in\mathcal{N}}$
  Set $\tau = \tau + 1$
  Realize disturbances
  System transitions to the new state

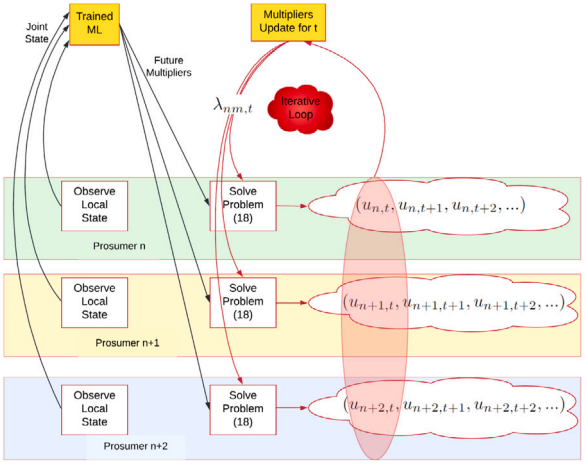


Fig. 2. Online procedure at a timeslot t.

s.t. (1)–(6), (8),

$$p^{\mathrm{g}}_{i,t}, p^{\mathrm{f}}_{i,t}, p_{ij,t} = p^{\mathrm{g}}_{i,t}[k-1], p^{\mathrm{f}}_{i,t}[k-1], p_{ij,t}[k-1], \quad \forall i \neq n, j \in \mathcal{C}_i, t \in \mathcal{T},$$

$$p^{\mathrm{g}}_{n,t}, p^{\mathrm{f}}_{d,t}, p_{nm,t} = \widetilde{p}^{\mathrm{g}}_{n,t}, \widetilde{p}^{\mathrm{f}}_{n,t}, \widetilde{p}_{nm,t}, \quad \forall t < \tau, n, m \in \mathcal{N}.$$

The distributed point-forecast MPC algorithm is described in Algorithm 1.

The shortcoming of this method is that it uses a single point-forecast (scenario) for future disturbances, which makes it ignorant to the full statistical information available. One could enhance the method's performance by generating multiple scenarios for future disturbances and replace problem (19) with a stochastic program. However, due to the multi-dimensional uncertainties of the problem, a large number of scenarios is required, which quickly becomes problematic, since the number of variables of the stochastic program grows proportionally to the number of scenarios. In the next subsection, we present the proposed policy which side-steps this issue.

3.3. Learning-based, distributed model predictive control

In an ML-assisted approach, we solve multiple instances (scenarios) of the deterministic optimization problem (13) offline, for each one obtaining the sequence of realized states $[\boldsymbol{x}^s_t]_{t\in\mathcal{T}}$ and corresponding optimal control actions $[\boldsymbol{u}^s_t]_{t\in\mathcal{T}}$. By generating a set $\mathcal{Y}$ of multiple pairs $y \in \mathcal{Y}$ of the form

$$y \triangleq (\boldsymbol{x}^s_t, \boldsymbol{u}^s_t), \tag{20}$$

we can, in theory, train a machine-learning module to return the optimal action $\boldsymbol{u}_\tau$ once presented with the current state $\boldsymbol{x}_\tau$ at current timeslot τ of the online operation. In this way, the computational burden, discussed in the previous subsection, is outsourced to the offline procedure, i.e. the training phase, and the online decision is made by the trained ML algorithm in a matter of a few seconds (at most). This is also called the "learn-to-optimize" paradigm.

This approach, however, creates two major challenges: first, the resulting controls must respect the power balance Eqs. (6), (7) (which the ML algorithm does not generally guarantee) and, second, the dimensionality of the action space renders the learning unrealistically data-intensive (recall also that in the learn-to-optimize approach the data is not readily available, but to obtain $|\mathcal{T}|$ pairs y, we need to solve an optimization problem, namely (13)).

Notice, however, that problem (13) is a convex optimization problem, featuring a set of optimal dual variables $\lambda_{nm,t}$ corresponding to constraints (7) and observe that, when solving problem (13) for a scenario s, the optimal duals $\lambda^s_{nm,t}$ can also be attained. Our proposition is to use the scenarios and corresponding instances of problem (13) to obtain pairs of the form

$$y^* \triangleq (\boldsymbol{x}^s_t, (\lambda^s_{nm,\hat{t}})_{n\in\mathcal{N},m\in\mathcal{C}_n,\hat{t}\in[t+1,|\mathcal{T}|]}), \tag{21}$$

i.e., connect an occurring state with the optimal *dual* variables of the subsequent timeslots, instead of the optimal control (primal) variables. This twist effectively resolves both of the challenges mentioned. Beginning from the second challenge, observe that when learning the optimal duals instead of the primals, the learning task's dimension (and therefore the necessary data) is dramatically reduced, since the dual variables are only $\frac{1}{2}|\mathcal{N}|\cdot|\mathcal{C}_n|\cdot|\mathcal{T}|$ (in contrast to the number $|\mathcal{N}|\cdot|\mathcal{T}| + |\mathcal{D}_n|\cdot|\mathcal{N}|\cdot|\mathcal{T}| + |\mathcal{N}|\cdot|\mathcal{C}_n|\cdot|\mathcal{T}|$ of primal variables). Thus, a ML algorithm can be trained to return the (predicted) optimal duals of the timeslots ahead, once presented with the observed state $\boldsymbol{x}_\tau$ at timeslot τ of online operation. Learning the duals of an economic dispatch problem was also applied in [6], although only for reducing the number of iterations of ADMM in a deterministic setting and without p2p transactions. In our context, we use the predicted duals in online operation as part of a direct look-ahead policy, an early (simpler) version of which was first proposed in [30] for a different problem.

Specifically, at iteration k and current timeslot τ, each prosumer $n \in \mathcal{N}$ solves the local optimization problem (19), similarly to the MPC policy of the previous subsection, effectively addressing the challenge of local constraint satisfaction (since each prosumer takes care of them when solving problem (19)). In our direct lookahead policy, when executing the ADMM procedure we update only the multipliers $\lambda_{nm,\tau}$ of the current timeslot, while the multipliers $(\lambda_{nm,t})_{t>\tau}$ of future timeslots are fixed to the values supplied by the ML module. This enhances the policy's performance since the predicted optimal future multipliers encapsulate relevant information about future expected demand arrivals, without the need to run multiple scenarios in online operation. The online procedure, at a timeslot t, is graphically illustrated in Fig. 2.

After the ADMM converges, the controls for the current timeslot are implemented and the system transitions to the next timeslot $\tau+1$ where the new state is communicated to the ML and the ADMM procedure is repeated using the new multipliers supplied by the ML for the future timeslots $t > \tau + 1$. The exact algorithm is described in Algorithm 2. The next subsection instantiates the specific ML technique used for the learning task described.

3.4. Learning the duals

In this subsection we elaborate on the ML algorithm developed, towards learning to associate a system's observed state $\boldsymbol{x}_\tau$ at current timeslot τ, to the optimal dual variables $\left(\lambda_{nm,t}\right)_{t\in\mathcal{T}:t>\tau}$ corresponding to

Algorithm 2 Model predictive control with direct lookahead policy

1: Initialize $\tau = 0$
2: **While** $\tau \in \mathcal{T}$
3: Observe system state $\boldsymbol{x}_\tau$
4: Feed state $\boldsymbol{x}_\tau$ to the ML
5: Set the future multipliers $(\lambda_{nm,t})_{t>\tau,n\in\mathcal{N},m\in\mathcal{C}_n}$ as prescribed by the ML
6: Initialize $k = 1$, $(\lambda_{nm,\tau}[0])_{n,m\in\mathcal{N}} = 0$
7: Initialize $(p^{\mathrm{g}}_{n,t}[0], p^{\mathrm{f}}_{d,t}[0], p_{nm,t}[0])_{t\in\mathcal{T}}$ at random
8: **while** $\max_{n\in\mathcal{N},m\in\mathcal{C}_n}\{p_{nm,\tau}[k] + p_{mn,\tau}[k]\} > \varepsilon$:
9: Each prosumer solves problem (19) to decide $p^{\mathrm{g}}_{n,t}[k], p^{\mathrm{f}}_{d,t}[k], p_{nm,t}[k]$
10: Set $\lambda_{nm,\tau}[k+1]$ using (16)
11: Set $k = k + 1$
12: Set $(\widetilde{p}^{\mathrm{g}}_{n,\tau}, \widetilde{p}^{\mathrm{f}}_{n,\tau}, \widetilde{p}_{nm,\tau})_{n,m\in\mathcal{N}}$ to the converged values $(p^{\mathrm{g}}_{n,\tau}[k-1], p^{\mathrm{f}}_{d,\tau}[k-1], p_{nm,\tau}[k-1])_{n,m\in\mathcal{N}}$
13: Set $\tau = \tau + 1$
14: Realize disturbances
15: System transitions to the new state

future timeslots. This refers to the ML used in line 5 of Algorithm 2. To this end, a Neural Network (NN) is utilized.

We opt for a simple NN with five dense layers. At each layer we perform a batch normalization in order to standardize the input data. This way, the learning process would be stabilized and the training epochs required are reduced. As a means to prevent overfitting, we added a dropout layer after each normalization layer. The dimension of the first layer equals the system's state dimension, while the final, output layer has the dimension of the Lagrange Multipliers of all prosumer connections in time, $\lambda_{nm,t}$. The ReLU activation function was used for each node of the NN. In addition, due to many demands not being present at a given timeslot, the state vector $\boldsymbol{x}_t$ contains many null (zero) values that complicate the model's learning. In order to confront this difficulty, we added a masking layer before the NN's input. This layer deactivates the zero value neurons of the input and thus they do not take part in the learning process. Finally, during the training phase we added an extra condition which is called early stopping. In this technique, the training stops when the difference between the previous epoch loss and current epoch loss is below a threshold for 3 consecutive timeslots.

4. Performance evaluation

4.1. Benchmarks

To assess the performance of the proposed algorithm (Algorithm 2), we compare it with two benchmarks. The first benchmark is the optimal-in-hindsight solution of the system. This could be obtained, only in theory, if all the information about the future (namely, future demand arrivals and their exact characteristics) was known beforehand. In that case, the optimal set of market exchanges would correspond to the solution of the deterministic optimization problem:

$$\min_{p^{\mathrm{g}}_{n,t}, p^{\mathrm{f}}_{d,t}, p_{nm,t}} \left\{ \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \left(c^{\mathrm{g}}_{n,t} + \sum_{d\in\mathcal{D}_n} c^{\mathrm{d}}_{d,t} + c^{\mathrm{ex}}_{n,t} \right) \right\}$$
$$\text{s.t. (1)–(8)}, \tag{22}$$

which is a convex program that can be efficiently solved using off-the-self solvers.

The second benchmark is a plausible conservative policy, where each demand is served upon arrival without looking at the current prices or reasoning about future prices. Notably, this is the current practice in actual power systems, where the demand is considered inflexible and is served immediately upon arrival, optimizing only the supply side so as to meet the demand at minimum generation cost. In our model, the conservative benchmark is obtained by fixing the consumption $p^{\mathrm{f}}_{d,t}$ of each present demand at

$$p^{\mathrm{f}}_{d,t} = \min\left\{ \overline{\mathrm{p}}_d, \left[\mathrm{E}_d - \sum_{\tau\in[\mathrm{t}^{\mathrm{a}}_d, t-1]} p^{\mathrm{f}}_{d,\tau} \right]^+ \right\}, \tag{23}$$

where $[x]^+$ denotes the operation $\max\{0, x\}$, and then solving only for the optimal generation and exchanges of the present timeslot, as in

$$\min_{p^{\mathrm{g}}_{n,t}, p^{\mathrm{f}}_{d,t}, p_{nm,t}} \left\{ \sum_{n\in\mathcal{N}} \left(c^{\mathrm{g}}_{n,t} + \sum_{d\in\mathcal{D}_n} c^{\mathrm{d}}_{d,t} + c^{\mathrm{ex}}_{n,t} \right) \right\}$$
$$\text{s.t. (1)–(8), (23)}. \tag{24}$$

The process (solving problem (24)) is repeated at each timeslot with the updated demands.

4.2. Evaluation setup

The proposed scheme and the two benchmarks were tested in a system with 12 prosumers of which six were consumers and the other six were producers during all 24 timeslots of each experiment.[4] During all experiments the prosumers connection graph was as shown in Fig. 3, while the values of the model's cost parameters were set as described in Table 2.

Each consumer was assumed to make one demand over the 24 timeslots and the arrival and departure times for each scenario were sampled by a Poisson distribution with a Poisson factor of 5 and 15 respectively. The demands' energy requirements were calculated as a function of $\mathrm{t}^{\mathrm{a}}_d, \widetilde{\mathrm{t}}^{\mathrm{d}}_d, \overline{\mathrm{p}}_d$ in order to ensure the problem's feasibility.

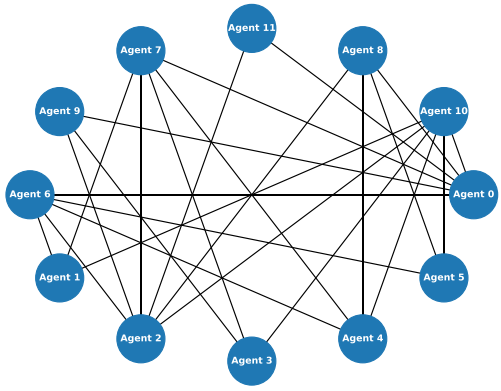


Fig. 3. Graph of prosumers' bilateral connections.

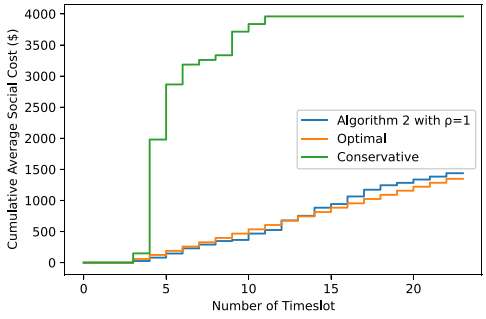
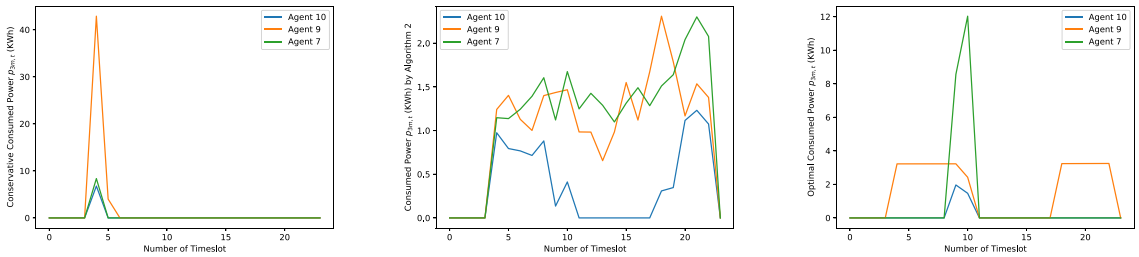


Fig. 4. Average system cost accumulated by each scheme over the horizon.

[4] Such a small number of prosumers makes the system's dual variables more volatile and harder to predict since they are sensitive to the state of each individual prosumer. Thus, such a setup is considered as a more challenging for the proposed algorithm.



(a) Power exchanges of peer 3, under the conservative policy

(b) Power exchanges of peer 3, under the proposed policy

(c) Optimal-in-hindsight power exchanges of peer 3.

Fig. 5. Power exchanges of peer 3 with its neighbors, under the three schemes.

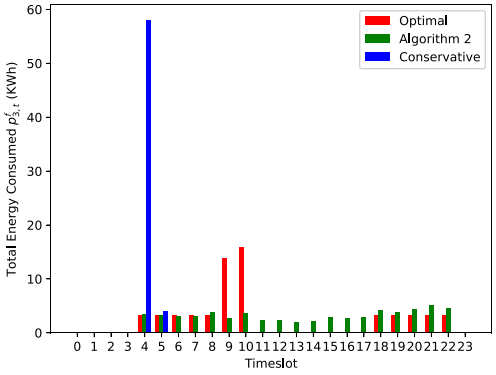


Fig. 6. Total Energy Consumed by Prosumer 3 in each timeslot.

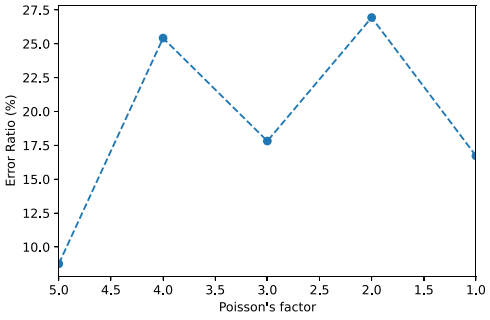


Fig. 7. Error ratio as a function of the error in the Poisson factor.

Table 2
Parameter values.

Prosumer	a_n	b_n	q_{mn}	$\underline{p}_{nm}$	$\overline{p}_{nm}$
Agent 0	0	0	0	−34	0
Agent 1	0	0	0	−54	0
Agent 2	0	0	0	−47	0
Agent 3	0	0	0	−58	0
Agent 4	0	0	0	−50	0
Agent 5	0	0	0	−49	0
Agent 6	0.3351	0.3297	0.9326	0	52
Agent 7	0.0498	0.4701	0.6676	0	36
Agent 8	0.8167	0.4464	0.4040	0	56
Agent 9	0.2919	0.6128	0.6654	0	51
Agent 10	0.4254	0.2279	0.4208	0	50
Agent 11	0.5444	0.4998	0.9972	0	57

4.3. Simulation results

The NN was trained using 2400 problem instances and tested in another 240 instances. The MAE of tested values was 0.45. The results to be presented were averaged out over a number of experiments.

Fig. 4 presents the comparison of the three schemes with respect to the average (over experiments) system cost c_t, as defined by Eq. (10), that they accumulate over the horizon's timeslots. As can be observed, the conservative policy accumulates cost early on, since it serves the demands immediately upon arrival. Consequently, it does not suffer any cost after a certain point, but the total cost accumulated by the end of the horizon is by far greater than the one of the proposed method. In contrast, the proposed algorithm refrains from serving the demands early (although a little bit too much compared to the optimal-in-hindsight solution). This results in a delayed accumulation of cost, stemming also from the disutility suffered by some demands, based on Eq. (4). Overall, looking at the total cost accumulated at the end of the horizon, it becomes apparent that the proposed method achieves a cost quite close to the theoretical optimal-in-hindsight solution, exhibiting a dramatic improvement over the conservative benchmark. In fact, the accumulated cost of the proposed solution is in the order of 8.5% higher than the one of the optimal-in-hindsight solution.

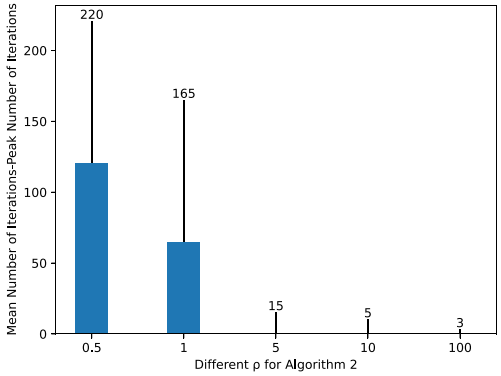
To gain insight into these results, we present the resulted power exchanges of a certain peer (peer 3) for each of the three schemes in Fig. 5 and the total energy consumed by peer 3 across the horizon in Fig. 6.

Note, however, that this result is achieved by assuming accurate statistical knowledge over the demands' characteristics (i.e. we do not know the characteristics of future demands but we do know their probability distributions, since those are used to generate the offline instances on which the NN was trained). To test the method's sensitivity/robustness against inaccurate knowledge of the demands' probability distributions, we use the trained model in a setup where the demands are generated from a Poisson distribution that is different than the one on which the model was trained. Specifically, the Poisson distribution of the demands' arrivals was modified in the test instances. Fig. 7 shows the proposed algorithm's error ratio as a function of the Poisson factor (with 5 being the factor on which the model was trained). The error ratio is defined as the percentage ratio of the average accumulated cost of the proposed algorithm (at the end of the horizon) to the one of the optimal-in-hindsight solution.

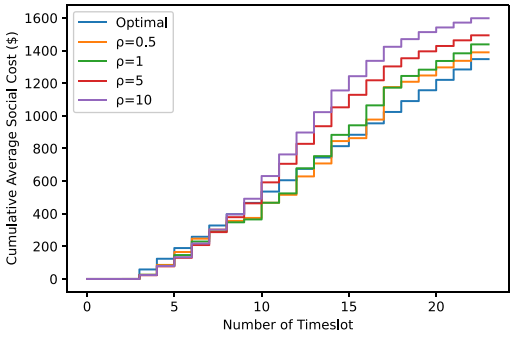
Finally, it is worth noticing that the parameter ρ of the ADMM procedure (built into the proposed algorithm) controls the trade-off between computational time and efficiency, i.e., a higher ρ makes the algorithm converge fast but sacrifices efficiency, while a smaller ρ achieves better results at the expense of more algorithm's iterations. These points are quantified in Fig. 8(a) and (b).

5. Conclusions

In this paper, we considered the problem of operating a p2p prosumer-centric energy market under uncertain energy demands. The problem was formulated as an optimal control problem with distributed networked agents. We presented a novel control policy that satisfies the local prosumers constraints by design, while exploiting statistical information about uncertainties in a scalable manner. The method



(a) Mean and peak number of iterations for different ρ



(b) Accumulated system cost for different values of ρ

Fig. 8. Effects of the choice of the ρ parameter of ADMM.

was experimentally shown to achieve a system cost close to the one of the (full information) optimal-in-hindsight solution, significantly outperforming a conservative benchmark policy. Future work should focus on further improving the proposed method in several directions. Specifically, privacy preserving methods, such as differential Privacy, should be integrated in the message exchange process, to ensure security in peers' communications. In addition, the training of neural networks could be done with robust techniques, to enhance the quality of predictions in a large input domain and minimize the convergence time for an out of the box input. Finally, one could test different Neural Networks architectures such as Graph Neural Networks, to boost the performance of the proposed framework.

CRediT authorship contribution statement

Georgios Tsaousoglou: Conceptualization, Methodology, Writing – original draft. **Petros Ellinas:** Software, Investigation, Data curation, Visualization. **Emmanouel Varvarigos:** Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

All data used were generated as described precisely within the manuscript.

Acknowledgments

This research was supported by Innovation Fund Denmark, through the project Flexible Energy Denmark (FED) (Grant No. 8090-00069B), and by the European Commission through projects ebalance+ (Grant No. 864283), and FLEXGRID (Grant No. 863876).

References

[1] Morstyn T, Farrell N, Darby SJ, McCulloch MD. Using peer-to-peer energy-trading platforms to incentivize prosumers to form federated power plants. Nature Energy 2018;3(2):94–101.
[2] Soto EA, Bosman LB, Wollega E, Leon-Salas WD. Peer-to-peer energy trading: A review of the literature. Appl Energy 2021;283:116268.
[3] Tushar W, Yuen C, Saha TK, Morstyn T, Chapman AC, Alam MJE, et al. Peer-to-peer energy systems for connected communities: A review of recent advances and emerging challenges. Appl Energy 2021;282:116131.
[4] Sorin E, Bobo L, Pinson P. Consensus-based approach to peer-to-peer electricity markets with product differentiation. IEEE Trans Power Syst 2019;34(2):994–1004. http://dx.doi.org/10.1109/TPWRS.2018.2872880.
[5] Baroche T, Moret F, Pinson P. Prosumer markets: A unified formulation. In: 2019 IEEE Milan PowerTech. IEEE; 2019, p. 1–6.
[6] Ruan G, Zhong H, Wang J, Xia Q, Kang C. Neural-network-based Lagrange multiplier selection for distributed demand response in smart grid. Appl Energy 2020;264:114636.
[7] Esmat A, de Vos M, Ghiassi-Farrokhfal Y, Palensky P, Epema D. A novel decentralized platform for peer-to-peer energy trading market with blockchain technology. Appl Energy 2021;282:116123.
[8] Gough M, Santos SF, Almeida A, Lotfi M, Javadi MS, Fitiwi DZ, et al. Blockchain-based transactive energy framework for connected virtual power plants. IEEE Trans Ind Appl 2021;58(1):986–95.
[9] Gough M, Santos SF, Almeida A, Javadi M, AlSkaif T, Castro R, et al. Development of a blockchain-based energy trading scheme for prosumers. In: 2021 IEEE Madrid PowerTech. 2021, p. 1–6. http://dx.doi.org/10.1109/PowerTech46648.2021.9494810.
[10] Cali U, Cakir O. Energy policy instruments for distributed ledger technology empowered peer-to-peer local energy markets. IEEE Access 2019;7:82888–900.
[11] Zhang C, Wu J, Long C, Cheng M. Review of existing peer-to-peer energy trading projects. Energy Procedia 2017;105:2563–8.
[12] Paudel A, Sampath LPMI, Yang J, Gooi HB. Peer-to-peer energy trading in smart grid considering power losses and network fees. IEEE Trans Smart Grid 2020;11(6):4727–37. http://dx.doi.org/10.1109/TSG.2020.2997956.
[13] Khorasany M, Mishra Y, Ledwich G. A decentralized bilateral energy trading system for peer-to-peer electricity markets. IEEE Trans Ind Electron 2019;67(6):4646–57.
[14] Jing R, Xie MN, Wang FX, Chen LX. Fair P2P energy trading between residential and commercial multi-energy systems enabling integrated demand-side management. Appl Energy 2020;262:114551.
[15] Ferro G, Robba M, Haider R, Annaswamy AM. A distributed optimization based architecture for management of interconnected energy hubs. IEEE Trans Control Netw Syst 2022.
[16] Javadi MS, Esmaeel Nezhad A, Jordehi AR, Gough M, Santos SF, Catalão JP. Transactive energy framework in multi-carrier energy hubs: A fully decentralized model. Energy 2022;238:121717. http://dx.doi.org/10.1016/j.energy.2021.121717, URL https://www.sciencedirect.com/science/article/pii/S0360544221019654.
[17] Xia B, Shakkottai S, Subramanian V. Small-scale markets for a bilateral energy sharing economy. IEEE Trans Control Netw Syst 2019;6(3):1026–37.
[18] Le Cadre H, Jacquot P, Wan C, Alasseur C. Peer-to-peer electricity market analysis: From variational to generalized Nash equilibrium. European J Oper Res 2020;282(2):753–71.
[19] Belgioioso G, Ananduta W, Grammatico S, Ocampo-Martinez C. Energy management and peer-to-peer trading in future smart grids: a distributed game-theoretic approach. In: 2020 European control conference. ECC, IEEE; 2020, p. 1324–9.
[20] Belgioioso G, Ananduta W, Grammatico S, Ocampo-Martinez C. Operationally-safe peer-to-peer energy trading in distribution grids: A game-theoretic market-clearing mechanism. IEEE Trans Smart Grid 2022.
[21] Tsaousoglou G, Giraldo JS, Paterakis NG. Market Mechanisms for Local Electricity Markets: A review of models, solution concepts and algorithmic techniques. Renew Sustain Energy Rev 2022;156:111890.
[22] Vahedipour-Dahraie M, Rashidizadeh-Kermani H, Shafie-Khah M, Siano P. Peer-to-peer energy trading between wind power producer and demand response aggregators for scheduling joint energy and reserve. IEEE Syst J 2020;15(1):705–14.
[23] Rashidizadeh-Kermani H, Vahedipour-Dahraie M, Shafie-khah M, Siano P. A peer-to-peer energy trading framework for wind power producers with load serving entities in retailing layer. IEEE Syst J 2021.

[24] Guo Z, Pinson P, Chen S, Yang Q, Yang Z. Online optimization for real-time peer-to-peer electricity market mechanisms. IEEE Trans Smart Grid 2021;12(5):4151–63.
[25] Morstyn T, McCulloch MD. Multiclass energy management for peer-to-peer energy trading driven by prosumer preferences. IEEE Trans Power Syst 2019;34(5):4005–14. http://dx.doi.org/10.1109/TPWRS.2018.2834472.
[26] Monasterios PRB, Verba N, Morris E, Morstyn T, Konstantopoulos G, Gaura E, et al. Incorporating forecasting and peer-to-peer negotiation frameworks into a distributed model predictive control approach for meshed electric networks. IEEE Trans Control Netw Syst 2022.
[27] Sun H, Chen X, Shi Q, Hong M, Fu X, Sidiropoulos ND. Learning to optimize: Training deep neural networks for interference management. IEEE Trans Signal Process 2018;66(20):5438–53.
[28] Powell WB. A unified framework for stochastic optimization. European J Oper Res 2019;275(3):795–821.
[29] Bertsekas D. Dynamic programming and optimal control: Volume I. vol. 1, Athena scientific; 2012.
[30] Tsaousoglou G, Mitropoulou K, Steriotis K, Paterakis NG, Pinson P, Varvarigos E. Managing distributed flexibility under uncertainty by combining deep learning with duality. IEEE Trans Sustain Energy 2021;12(4):2195–204.