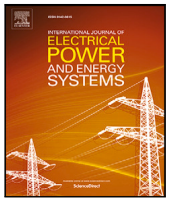
Electrical Power and Energy Systems 155 (2024) 109604



Contents lists available at ScienceDirect

International Journal of Electrical Power and Energy Systems

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





Residential flexibility characterization and trading using secure Multiparty Computation

Fairouz Zobiri [a,*], Mariana Gama [b], Svetla Nikova [b], Geert Deconinck [a]

[a] Department of Electrical Engineering (ESAT-ELECTA & EnergyVille), KU Leuven, 10 Kasteelpark Arenberg, Heverlee, 3001, Belgium
[b] Department of Electrical Engineering (imec-COSIC), KU Leuven, 10 Kesteelpark Arenberg, Heverlee, 3001, Belgium

ARTICLE INFO

Keywords:
Demand response
Privacy-by-design
Secure multiparty computation
Double auction
Receding horizon control

ABSTRACT

The intermittent nature of renewable energy sources challenges the ability of grid operators to maintain balance between supply and demand of electricity. Demand response, or flexibility, is the ability of a load to modify its consumption in response to a control signal. The necessary two-way communications between end-users and grid operators introduce new risks on consumer privacy and data security. Consumer engagement in such programs tends to be low not only for privacy issues, but also because of the inappropriate incentive schemes. In this work, secure Multiparty Computation is used in a flexibility market where end-consumers provide demand response to grid operators in exchange for a monetary compensation. Multiparty computation is a cryptographic method that performs operations over encrypted data. Users submit their offers in encrypted form, and a novel way to characterize flexibility as a commodity and to determine device constraints with very little information from the consumers is developed. The market is cleared via a double auction-like process. The method is validated using flexibility offers from 200 devices of different types. We demonstrate the possibility to clear the market and allocate resources with encrypted orders for sole information.

1. Introduction

The increased electrification of every aspect of human life, in order to meet environmental objectives, is putting an unprecedented strain on power systems. On the one hand, sectors such as transportation and heating, currently powered by fossil fuels, are shifting toward electricity-based solutions via electric vehicles (EVs) and heat pumps [1]. On the other hand, the supply-side is increasingly reliant on intermittent renewable energy sources (RES). The non-dispatchable nature of RES coupled with the steady increase in the load is pushing the grid to extremes, thus calling for a change in power system management.

Demand response (DR) is one of the recent novelties in grid control methods. It has been enabled by the emergence of the smart grid and smart devices, by allowing the demand side to take a more active role in the balance of demand and supply. It is usually defined as the ability of consumers to reshape their load profile in response to a price signal or an incentive [2]. These programs are currently restricted to large industrial consumers, whereas households and smaller businesses, represent an important untapped source of DR.

Despite recent studies on the potential of residential DR to take part in wholesale markets [3,4], or to offer ancillary services [5,6], several challenges still hinder the use of residential demand response. As a change in consumption of small consumers is unlikely to produce a change in grid conditions, the DR contribution of a large number of flexible devices needs to be aggregated in order to produce a significant impact. Direct control of devices runs into scalability issues, further diminishing the interest of grid operators in residential DR. On the consumers side, the reticence toward DR programs stems from the cost-benefit ratio that is oftentimes unbalanced. Residential users are usually part of a DR aggregator's portfolio. Aggregators may opt for a price-based scheme with the promise of lower energy bills, or an incentive-based scheme where a fraction of the revenues are shared with the end-consumer. In either model, the consumers do not get to set a value for their own flexibility that compensates their loss of comfort. Moreover, in many DR programs, the aggregator requests access to the full consumption profile of the household. This is often seen by end-users as a violation of privacy. Several studies [7] have shown that many consumer habits can be deduced from granular load profiles. This legitimizes the fears of consumers about the malicious use of their data.

One possible solution to the scalability issues is to create local DR markets to relieve the congestions on localized zones of the low voltage (LV) grid, or help balance supply and demand during episodes of low RES reliability. Localized markets would be small enough for consumers to put a value on their own flexible devices. Such markets

* Corresponding author.
E-mail address: fairouz.zobiri@kuleuven.be (F. Zobiri).

https://doi.org/10.1016/j.ijepes.2023.109604
Received 16 May 2023; Received in revised form 21 August 2023; Accepted 17 October 2023
Available online 31 October 2023
0142-0615/© 2023 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

can also be secured via privacy-preserving cryptographic techniques, that would not be applicable to large national/international markets. In this work, we introduce a novel local DR market model. It is run by a trading platform (TP), and puts end-consumers (connected to the LV grid) in contact with DR buyers. The buyers who bid in such a market can be system operators (SO) acquiring flexibility for congestion management or voltage regulation. They can also be balance responsible parties (BRP) or suppliers aiming at balancing their portfolio to avoid imbalance penalties. Both parties freely select the bids and offers to submit to the DR market. Individual offers are secured via encryption and never revealed to a third-party. The market is cleared via a double auction-like mechanism, where calculations are performed via a computation-over-encrypted-data method called secure MultiParty Computation (sMPC).

1.1. Literature review

The available literature combining end-user privacy and demand response programs, or smart grid applications in general, can be classified into three categories: decentralized DR methods, distributed ledger technologies (DLT), and cryptographic methods.

Decentralized algorithms differ from centralized control algorithms in terms of where the decisions are made [8]. In centralized methods, a central controller orchestrates the entire system. In decentralized methods, each participant computes its own control policy locally, thus avoiding external share of its data. The original purpose of decentralized DR programs [8–13] was to address scalability problems due to the growing number of participants and the complexity of the smart grid. Lately, the use of decentralization to achieve privacy as well has gained momentum during the last years [10,11]. Several methods can be used to achieve decentralization. The Alternating Direction Method of Multipliers (ADMM) is used in [10] to solve an optimal power flow in the presence of DR, in order to avoid disclosing financial information. Multi-agent control theory is used in [9] where consensus control of multi-agent systems is deployed over a directed communication network to build a decentralized DR program. The same technique is used in [12], where each agent learns an optimal policy using Distributed W-Learning, a form of reinforcement learning. Another class of algorithms relies on game-theoretic methods. For instance, in [13], a non-cooperative game is used to model an energy sharing DR program.

The second class of algorithms involves blockchain-based DLT to achieve a form of decentralization. Permissioned blockchain is often preferred over permissionless for increased privacy. In [14], DR is enforced via a peer-to-peer trading framework between electric vehicles (EVs) and service providers in a vehicle-to-grid (V2G) setting. A double auction is carried out over a permissioned consortium blockchain to allocate flexibility. Some studies [15,16] prefer the storage of metering electricity data over the blockchain. Smart contracts then determine the baseline and the ability of each user to modify their load schedule. As blockchains alone often are not enough to ensure privacy, some authors add anonymity schemes to their approach. For example, in [16] a digital pseudo-identity is assigned to the participants of a DR program for valley filling and peak shaving, while in [17], a mutual authentication scheme between the smart meter and the DR control unit is required prior to any exchange of sensitive data. Finally, in [18] the GUARDIAN blockchain scheme is introduced. In this scheme end-users initialize transactions by creating a new block. A set of miner nodes proceeds to verify the blocks, and only then transactions can take place.

In cryptographic methods, cryptographic primitives are used to secure part or the whole of the DR scheme [19–26]. In such methods, anonymity of end-users is often employed to maintain privacy [19,22]. In [19], the authors propose an approach where the DR service provider analyzes the fine-grained metering data to reschedule the demand, while the customer identity remains private through a combination of identity-committable signatures [27], zero-knowledge proofs [28], and partially blind signatures [29]. In [24], a social welfare maximization problem is solved through homomorphic encryption in a program where EVs provide DR during microgrid outages. The authors in [22] build a protocol around openADR [30] that guarantees the security of the bidders from several aspects such as anonymity, data privacy, or non-impersonation. The protocol relies on El-Gamal public key encryption [31] and Shnorr's signature [32] to achieve the set of goals.

Despite the diversity of the above mentioned methods, a few gaps still remain in the treatment of privacy in DR programs. While decentralized methods ensure scalability and maintain user privacy, they lack the controllability and optimal nature of centralized methods. Additionally, reaching consensus still requires data exchange. As to Blockchain methods, even the permissioned ones, they require access to full consumer data sets, which despite anonymization, can still be used by third-parties to learn the behavior of individual users. Additionally, blockchain users have access to each other's data. Finally, the literature on cryptographic methods focuses on the communication protocols and often ignores the specificity of flexibility which unlike other traded commodities, involves extra constraints related to the devices' operational limits and the comfort level of the consumers.

1.2. Contribution

In this paper, we partially build on the work described in [33] to develop a private-by-design flexibility market. Market participants are only required to exchange encrypted offers with a central trading platform, rendering it impossible for any entity to learn user behavior. Individual consumer data is handled via sMPC, a cryptographic method that allows a set of mistrustful parties to jointly compute the output of a function without revealing the input.

Additionally, this work sets itself apart from other cryptography-based solutions by describing a way to characterize flexibility at market level from little information received from participants. This method does not require access to users' granular consumption data, but only minimal information about their devices. Such information includes power rating, completion time, as well as the time interval of availability. It is also simple enough to be supported by sMPC.

Once it characterizes flexibility, the central TP aggregates supply and demand. At this point, as data is aggregated and anonymous, it can be revealed and used in a double auction-like mechanism. The problem is formulated as a linear program, whose objective is optimal flexibility allocation and whose constraints are the aggregate comfort levels. The resulting optimal schedule is dispatched to the participants using sMPC again to avoid learning individual consumption schedules. The main contributions are summarized as follows:

- Private-by-design, incentive-based flexibility market that uses sMPC for data protection.
- A corresponding procedure to characterize flexibility from minimal information. The procedure is also simple enough to be carried out over encrypted data.
- An aggregate-dispatch auction mechanism inspired by [34], and extended to all types of devices, instead of only EVs as in [34].
- The remuneration scheme does not require knowledge of the consumers' baseline consumption, as the devices are directly scheduled by the TP.

This paper is organized as follows. In Section 2, a brief introduction to sMPC is given. Section 3 provides an overview of the market layout. Section 4 contains a detailed procedure for flexibility characterization for each type of device, and the computation of the set of constraints that combines device limitations and user comfort levels. In Section 5, the social welfare maximization problem is described, along with the final allocation mechanism. In Section 6, we provide a numerical example to demonstrate the validity of our approach. Section 7 draws the relevant conclusions from our work.

2. Secure multiparty computation

Secure Multiparty Computation or sMPC is a cryptographic technique that allows a set of mistrustful parties to jointly perform an operation on secret data. The parties, also referred to as players, have to make sure that only the output of the operation is known, and that the result of the operation is correct [35]. The potential of sMPC for achieving privacy-preserving auctions has motivated several studies. The first large-scale real-world application of sMPC, for the Danish sugar-beet contracts, was presented in [36]. This technique has also been considered for tackling privacy issues in the electricity sector. Studies ranging from smart meter data collection [37] to peer-to-peer electricity trading [38,39] have used sMPC.

In our market configuration, sMPC is used to handle the data of individual DR providers and buyers. It is used prior to the auction to aggregate the bids and offers of the market participants, and after the auction to allocate the results of the auction to individual participants. Due to the properties of sMPC, all individual data remains private, and even the parties executing the sMPC protocol cannot access it. However, the security provided by sMPC is only ensured if not all sMPC computing parties are maliciously colluding. To prevent this, computing parties are chosen to be entities with conflicting interests such that they have no benefit in colluding.

For individual participants to split their data among the computing parties, they rely on the principle of secret sharing. The simplest form of secret sharing is additive secret sharing. The authors of [40] describe this tool by first selecting a set of integers $\mathbb{Z}_p = \{0, 1, \dots, p-1\}$, such that p is a prime number. If we want to secretly share a number X with three computing parties, we generate three uniformly-distributed random numbers $x_1, x_2, x_3 \in \mathbb{Z}_p$, referred to as shares, such that $X = x_1 + x_2 + x_3 \mod p$, where $\mod$ is the modulo operation. We then send each share to one of the computing parties, which has to keep the share a secret.

As an example of how sMPC performs an operation, let us consider two users, with market orders with volumes of flexibility X and Y, respectively, and for which we want to compute the sum $S = X + Y$. Each user generates secret shares for their bid such that $X = x_1 + x_2 + x_3 \mod p$ and $Y = y_1 + y_2 + y_3 \mod p$. If we assume that there are three computing parties, party 1 receives x_1 and y_1, party 2 receives x_2 and y_2, while party 3 receives x_3 and y_3. Party 1 then sums the shares in their possession $s_1 = x_1 + y_1$, party 2 does $s_2 = x_2 + y_2$, and party 3 does $s_3 = x_3 + y_3$. Afterwards, the three parties exchange the partial sums with one another, e.g., party 1 sends s_1 to party 2 and party 3, and so on. In the end, each party will have s_1, s_2, s_3, and can compute the total sum $S = s_1 + s_2 + s_3$. Note that no party has knowledge of all three secret shares of either X or Y.

Another cryptographic technique commonly used for similar purposes as sMPC is homomorphic encryption (HE). However, HE introduces a significantly larger computational overhead than sMPC and is in general less efficient. In our protocol, using HE instead of sMPC would be prohibitively expensive. This means that the only way to use HE here would be to simplify our protocol, which would hinder the welfare maximization of the mechanism.

3. Market description

In this section, we lay down the basic rules governing the DR market, from the chronological order of operations to the structure of the bid and the type of devices considered. We describe the opening and closing times, the size of settlement periods, and expected delivery time of flexibility. The market order structure provides the template that users must follow so that their orders are taken into account. Finally, we list the types of device that can participate in the market.

3.1. Market characteristics

The market that we design in this work is a day-ahead (DA) market for demand response (DR), or flexibility, as the two terms are used here interchangeably. The market is run by a trading platform composed of three independent entities: a user representative, a TSO, and an aggregator. The three entities represent the computing parties in a sMPC framework. The market is a platform for end-consumers or prosumers, acting as DR providers, to meet potential flexibility buyers. The buyers are usually system operators (SO), balance responsible parties (BRP), or suppliers. The sellers are located in the same geographical area, but they do not have to be represented by the same supplier. They can also be part of the portfolio of one or several aggregators.

In this work, we consider that all the participating consumers/prosumers are equipped with a Home Energy Management System (HEMS). The user can enter each device model, and the HEMS (or the managing company) is responsible for extracting the device's specifications. This includes the time that the device needs to complete each task, and the corresponding maximum power rating for each setpoint.

The first few times, the user might need to enter the device's time limitation manually. After a while, the HEMS can learn the device's use patterns. There exist some studies in the literature that describe how to learn the flexibility of a device from a user's input data. We can mention [41], for example, where the authors learn the flexibility of a fleet of EVs based on historical charging sessions. Therefore, user involvement is kept to a minimum.

The market opens at 8 am and closes at 12 pm on the day preceding the expected flexibility activation. During this interval, market participants can submit their orders to buy or sell flexibility. The HEMS generated orders that cover settlement periods for the 24 h of the following day. Each settlement period is of length Δt in minutes, and can range from 15 to 60 min. The first trading period starts at 0h00, T_f is the number of time steps.

Prior to market opening, the TP publishes a price pool $\Lambda = \{\lambda_t^1, \dots, \lambda_t^K\}$, for all $t \in [0, T_f]$. The price pool can remain constant for the whole day, or the prices may vary with time, depending on the expected state of the network, and DR availability during the following day. Each market participant, seller or buyer, chooses a price from the pool and submits a sell offer or a buy bid to the market platform. Market orders follow the format presented in Section 3.2. As explained in Section 2, every information in an order is split into shares and each share is secretly sent to one of the computing parties composing the TP. Fig. 1 illustrates this process. The solid black arrows represent the share submission, while the red dashed arrows show that the notification process is done via shares as well.

After market closure, the TP models the flexibility of each device, using the information contained in the offer. It derives the power and energy constraints of each device. It then aggregates the flexibility within offers that have similar prices, obtaining a vector of K aggregate DR volumes, K being the number of price points. The buyers' bids are likewise aggregated. At each time t, power and energy constraints are all summed up together.

The TP then solves a social welfare maximization problem. It is formulated as a linear programming problem, composed of two parts, an auction-like term, and an optimal power allocation term. The latter term is used to find the optimal consumption schedule versus the DA electricity prices for the devices that were not selected, ensuring if not monetary compensation, a decrease in their energy bills.

In the last step, the quantities determined by the auction for the first settlement period are dispatched to the individual users, who are remunerated following a pay-as-bid scheme. Buyers are awarded flexibility proportionally to their bid volume. DR providers are divided into auction winners and consumers. The offers within each winning price bucket are ranked according to the urgency to consume, and the remaining time for which the appliance is available. The consumers are ranked similarly and the power is divided among them according

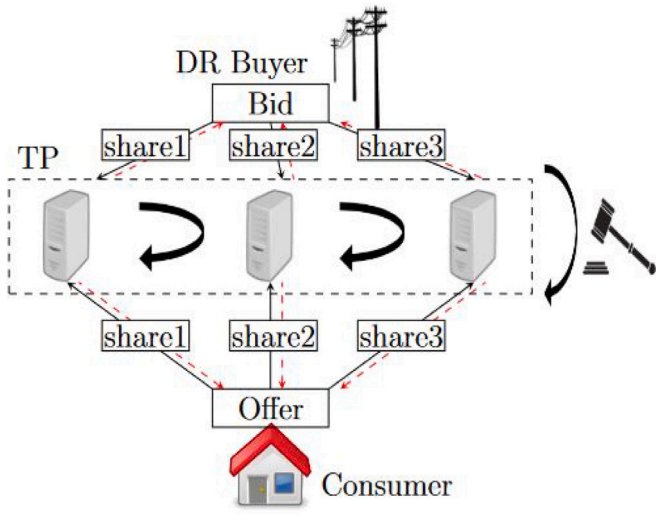


Fig. 1. Bid and offer submission process.

to their consumption needs. For privacy reasons, this step is carried out via sMPC.

After power dispatch, device constraints change as explained in Section 4. Therefore, the optimization horizon shifts by one time step to become $[1, T_f]$, and the optimization-dispatch process is repeated until the horizon $[T_f - 1, T_f]$.

3.2. Bid and offer structure

Trading flexibility as a commodity requires knowledge of the characteristics of the devices providing such service. This is paramount for respecting the ratings of the device and maintaining an acceptable level of user comfort, so as not to alienate consumers from further taking part in flexibility trading. However, due to the privacy requirements, and in addition to its encryption, the information transmitted in a market order is kept to a minimum.

Let N be the total number of flex offers submitted to the TP in a single day, and let $\mathcal{O}^i$ denote the offer of device i $(1 \leq i \leq N)$. The offer $\mathcal{O}^i$ should have the following structure

$$\mathcal{O}^i = \langle id^i, \overline{P}^i, \lambda^i, type^i, t^i_{min}, t^i_{max}, t^i_{run} \rangle,$$

where delimiters $\langle . \rangle$ indicate that the enclosed information is secretly shared. The enclosed quantities represent

- id: the consumer and device identifier,
- $\overline{P}$: maximum power that a device consumes in a cycle in kW,
- λ^i: ask price in €/kW. It is chosen from a pool $\{\lambda_1, \ldots, \lambda_K\}$ published by the TP prior to market opening,
- $type$: we separate the devices into three categories, introduced below,
- t_{min}: time at which the device becomes available in the flexibility market,
- t_{max}: deadline by which device cycle must be completed,
- t_{run}: running time of the device.

Note that we require that $t_{max} > t_{min} + t_{run}$, otherwise the device cannot be considered as flexible. Every market order represents a single device, and a consumer has the possibility to submit multiple offers for multiple devices at the same time. If, on a particular day, a consumer has no flexibility, they should at least submit one offer filled with zeros. This way, the TP cannot guess extra information from the number of participants. Market order generation is, however, outside the scope of this study which describes only the market level. For generating an order, the reader is referred to [42].

As buyers are not subjected to such device constraints, the buy bid follows a simpler format. The value of the imbalance varies from one time step to another, the buyers need to indicate their desired quantities for each time step. For the times at which they have no desire to buy, they bid a quantity of 0 kW. The buyers' bid follows the structure presented below, where $\mathcal{B}$ denotes a buy bid

$$\mathcal{B}^j = \langle id^j, I^j, \lambda^j, t \rangle,$$

- id^j is the identifier of buyer j,
- I^j is the quantity requested by buyer j. By convention, if $I^j > 0$, the buyer requires a decrease in consumption, also referred to as upward regulation. If $I^j < 0$, downward regulation, or an increase in consumption, is expected,
- λ^j is the price bid by buyer j, such that $\lambda_j \in \Lambda$,
- t is the time step

3.3. Device types

As there exists a large number of controllable devices that can offer DR in a flex market, the problem of scheduling the flexible devices can quickly become intractable. Therefore, as in we sort controllable into three categories, depending on the type of flexibility that they can offer, and the constraints that govern their operation. Several studies have used similar classifications, for instance, [43] refers to the categories as buckets, batteries and bakeries. However, we keep the same terminology as in [42]. The three categories are summarized below.

1. Thermostatically-Controlled Load (TCL). The power consumption of TCLs can be adjusted freely within a certain range, which usually corresponds to the temperature range considered comfortable by the users. TCLs consist of heating, ventilation, and air conditioning (HVAC) devices.
2. Batteries and Electric Vehicles (EV). Compared to the previous type, EVs offer more flexibility in time, up to their departure instant. The power that they receive can also vary continuously within a range $[0, \overline{P}]$. However, EVs need to be charged to a certain level by t_{max}.
3. Deferrable Load (DL). Similar to EVs, DLs have a high flexibility in time. The particularity of these devices is that they are uninterruptible once their cycle has started. They also have to follow a predefined power profile during their operation. DLs include washing machines, dishwashers, or dryers. For simplicity, we assume that the power profile is constant during a cycle and set to the maximum power $\overline{P}$ needed during a cycle.

4. Flexibility modeling

Throughout the literature, flexibility is often modeled as part of the demand or the supply in a wholesale market. Aggregated flexible assets are also often represented as a virtual battery that can charge, when power is consumed, or discharge, when the devices are either on standby or injecting power to the grid [6]. In this paper, we choose to model flexibility as a separate asset, different from supply and demand, traded in a market of its own. In such market, DR providers are referred to as sellers. Moreover, since all grid issues that require flexibility find their origin in a mismatch between supply and demand, the buy side in a flexibility market is referred to as the imbalance or regulation. This terminology is used to avoid any confusion with supply and demand in electricity markets.

In a DR market, two types of flexibility can be traded. Upward flexibility, or simply up flex, is defined as a decrease in demand at certain time and is used to compensate a lack of supply or congestion in the network. Downward flexibility (down flex), on the other hand, is an increase in consumption of a device. It can be provided, for instance, by charging a battery, or starting a wash cycle. It is used at times of

over generation to restore the balance in the network. Let $\overline{f}_t^i$ denote the upward flexibility of device $i \in 1, \dots, N$ at time step t, while $\underline{f}_t^i$ denotes its downward flexibility. For the sake of convention, we describe up flex with positive values, while down flex is a negative quantity. We describe below how flexibility is modeled for each type of device, as well as the associated power and energy constraints.

In what follows, to improve readability, we drop the encryption notation $\langle . \rangle$. The reader has to bear in mind that quantities bearing the superscript i or relating to individual bids are encrypted. Additionally, all the operations involving such quantities are performed using sMPC.

4.1. Thermostatically-controlled loads

TCLs can offer flexibility by deviating from their original set-point. The HEMS uses heat transfer equations to compute the time interval for which comfort levels can be maintained and the corresponding power deviation. It then transmits an offer indicating that a maximum power deviation $\overline{P}$ can be held for a duration $[t_{min}, t_{max}]$. Therefore, flex of a TCL i at time t can be expressed as

$$0 \leq \overline{f}_t^i \leq \overline{P}^i, \quad t \in [t_{min}, t_{max}], \tag{1a}$$

$$\underline{f}_t^i = 0. \tag{1b}$$

After t_{max}, the power goes back to its original setting, so that the output temperature remains within an acceptable range. Therefore, no energy constraints need to be associated with this type of devices. They are already embedded in t_{min} and t_{max}.

4.2. Electric vehicles

The energy stored in an EV battery can be described by the simplified model

$$E_{t+1}^i = E_t^i + P_t^i \Delta t, \tag{2}$$

where E_i^t is the energy stored in a battery i at time t, P_t^i is the power allocated to the battery at time t.

The HEMS submits t_{min}, the time at which the user is expected to plug in to the charging station, and t_{max}, the user's expected departure time. The charging rate is $\overline{P}$, and t_{run} is the time it takes to reach the desired state-of-charge (SoC) at this rate. The TP then knows that the battery needs to be charged up to $E_{lim}^i = \overline{P}^i t_{run}$, which is the amount of energy required to reach the desired SoC.

The battery has potential to offer both up and down flex at any given time step t. At $t = t_{min}$, the battery can offer upward flexibility from t_{min} up to $t_{max} - t_{run}$, at which point it has to start charging. If the battery is allocated some power or has offered down flex at a certain time $t \in [t_{min}, t_{max} - t_{run}]$ for a duration $\Delta\tau$, it subsequently can offer up flex for longer, i.e. until $t_{max} - t_{run} + \Delta\tau$. Fig. 2 illustrates this idea. The blue block indicates up flex, while the red block represents down flex. The filled area indicates that flex evolves continuously from 0 to $\overline{P}$ or $-\overline{P}$. Up and down flex of battery i at time t can be computed via the following equations

$$\underline{f}_t^i \in \begin{cases} [0, \overline{P}^i], & \text{for } t_{max} - t_{run} + \left\lceil \frac{E_t^i}{\overline{P}^i \Delta t} \right\rceil \leq t < t_{max} \\ & \text{and } E_t < E_{lim}^i, \\ \{0\}, & \text{otherwise,} \end{cases} \tag{3a}$$

$$\underline{f}_t^i \in \begin{cases} [-\overline{P}^i, 0], & \text{for } t_{max} - t_{run} + \left\lceil \frac{E_t^i}{\overline{P}^i} \right\rceil \leq t < t_{max} \\ & \text{and } E_t < E_{lim}^i, \\ \{0\}, & \text{otherwise,} \end{cases} \tag{3b}$$

where $\lceil . \rceil$ is the ceiling operation. As indicated in the offer, the power at which an EV battery can charge is comprised in the range $0 \leq P_t^i \leq \overline{P}^i$. The minimum and maximum energy stored in the battery at every time step (given by $\underline{E}_t^i$ and $\overline{E}_t^i$ respectively), when $t \in [t_{min}, t_{max})$, are given by

$$\underline{E}_t^i = \max(E_{t-1}^i, \overline{P}^i (t_{run}^i + t - t_{max}^i) \Delta t), \tag{4a}$$

$$\overline{E}_t^i = \min(\overline{P}^i (t - t_{min}^i) \Delta t, \ E_{lim}^i), \tag{4b}$$

Eq. (4b) indicates that the EV can start charging as soon as it is plugged in, but its urgency to consume depends on its deadline and the power previously received as per (4a).

4.3. Deferrable loads

DLs can provide up flex from time t_{min} to the time at which they must start $t_{max} - t_{run}$. Down flex, on the other hand, is available for the entire duration $[t_{min}, t_{max})$. However, if a DL is allocated power at a certain time step, even if $t < t_{max} - t_{run}$, it can no longer offer up flex, and must run until its cycle terminates. Fig. 3 illustrates this idea. The blue line represents up flex, while the red line indicates down flex. The absence of filling indicates that flex can only take the discrete values 0, $\overline{P}$, $-\overline{P}$, while t_1 indicates the time at which the device is started. The energy consumed by a DL at each settlement period is also described by (2), where the power allocated at each time step can only take on two values, i.e. $P_t^i \in \{0, \overline{P}^i\}$. Therefore, up and down flex of a DL in each period is computed as

$$\overline{f}_t^i \in \begin{cases} \{0, \overline{P}^i\}, & \text{for } t_{min} \leq t < t_{max} - t_{run} \\ & \text{and } P_{t-1}^i = 0, \\ \{0\}, & \text{for } t \notin [t_{min}, t_{max}) \text{ or } E_t^i = E_{lim}^i, \end{cases} \tag{5a}$$

$$\underline{f}_t^i \in \begin{cases} \{0, -\overline{P}^i\} & \text{for } t_{min} \leq t < t_{max} - t_{run}, \\ \{\overline{P}^i\} & \text{for } t_{max} - t_{run} \leq t < t_{max} \\ & \text{or } P_{t-1}^i > 0, \\ \{0\}, & \text{for } t \notin [t_{min}, t_{max}) \text{ or } E_t^i = E_{lim}^i, \end{cases} \tag{5b}$$

According to the characteristics of flexibility given above, the minimum and maximum total energy required by a DL at each time step, denoted likewise by $\underline{E}_t^i$ and $\overline{E}_t^i$, take the following values

$$\underline{E}_t^i = \begin{cases} 0, \text{if } E_{t-1}^i = 0 \text{ and } t_{min}^i \leq t < t_{max}^i - t_{run}^i, \\ E_{t-1}^i + \overline{P}^i \Delta t, \ \text{if} \ \ t \geq t_{max}^i - t_{run}^i \text{ or } E_{t-1}^i > 0, \end{cases} \tag{6a}$$

$$\overline{E}_t^i = \min(\overline{P}^i (t - t_{min}^i) \Delta t, \ E_{lim}^i). \tag{6b}$$

Eq. (6a) indicates that the minimum energy of a DL depends on time and power already consumed, while (6b) depends on time elapsed since t_{min} and the maximum energy E_{lim}^i.

5. Aggregation, auction and allocation

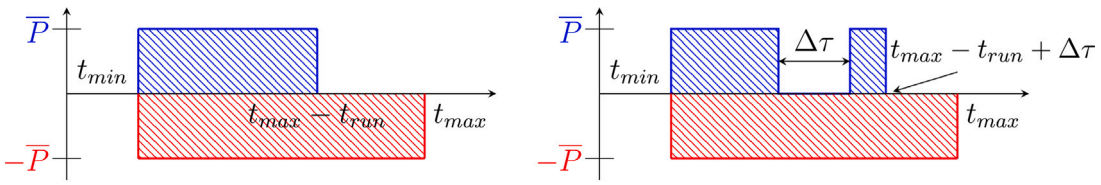
5.1. Aggregation

After computing the up and down flex available from each device, as well as their power and energy constraints, the TP aggregates flex at every price point, and all energy and power constraints. Let $\overline{F}_t^k$ and $\underline{F}_t^k$ denote the total up and down flex, respectively, offered at price λ_k at time t. Eqs. (7) and (8) describe the sum over encrypted data performed by the TP

$$\overline{F}_t^k = \sum_i \overline{f}_t^i |_{\lambda_i = \lambda_k}, \ \forall \ 1 \leq k \leq K, \tag{7}$$

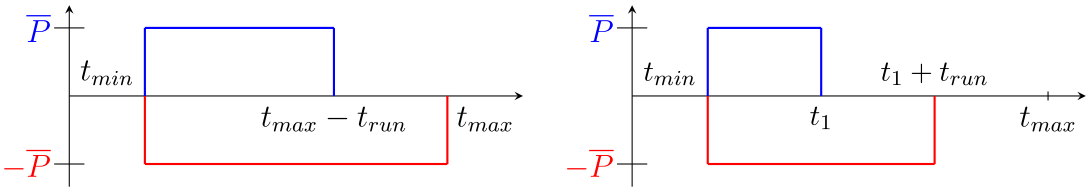
$$\underline{F}_t^k = \sum_j \underline{f}_t^j |_{\lambda_j = \lambda_k}, \ \forall \ 1 \leq k \leq K. \tag{8}$$

The sums in (7) and (8) are carried out over the secretly shared data $\overline{f}_t^i$ and $\underline{f}_t^i$. The result, however, can be revealed as it does not contain any information about the users' individual offers, or their identities.



(a) Initial flexibility (b) Evolution of flexibility

Fig. 2. Flexibility of battery-like devices.



(a) Initial flexibility (b) Possible evolution of flexibility

Fig. 3. Flexibility of deferrable devices.

Likewise, the TP aggregates the buyers' asking volumes at each price point. Let $\overline{I}_t^k$ and $\underline{I}_t^k$ denote the aggregated upward and downward regulations requested by the buyers at time t at price point k.

$$\overline{I}_t^k = \sum_i I_t^i|_{\lambda_i=\lambda_k,\ I_t^i>0},\ \forall\ 1 \le k \le K, \tag{9}$$

$$\underline{I}_t^k = \sum_j I_t^j|_{\lambda_j=\lambda_k,\ I_t^j<0},\ \forall\ 1 \le k \le K. \tag{10}$$

Using sMPC, the TP also aggregates the power and energy constraints of the flexible devices. There is no need to aggregate at every price point, even if it is possible. We denote the sum of the minimum and maximum energy, and maximum power at time t by E_t^{min}, E_t^{max}, and P_t^{max}, respectively. The sums are computed on encrypted data over all values $1 \le i \le N$, as shown in Eqs. (11)

$$P_t^{max} = \sum_{i=1}^{N} \overline{P}_t^i, \qquad E_t^{min} = \sum_{i=1}^{N} \underline{E}_t^i, \qquad E_t^{max} = \sum_{i=1}^{N} \overline{E}_t^i. \tag{11}$$

At this point, the aggregated power and energy constraints can also be revealed or decrypted. They will serve as the constraints of the double auction that allocates resources from the consumers to the buyers.

5.2. Auction

The TP has to run two simultaneous auctions, one for up flex and one for down flex. Both auctions are grouped in one linear programming problem. Its objective is to find the optimal resource allocation by selecting the buyers who are willing to pay the most, i.e., having higher wiliness to buy, and the sellers who bid the lowest prices, with higher wiliness to sell. This means that not all flex providers can win the auction. However, all the devices have power and energy constraints and need to be allocated a certain amount of power at each time step to satisfy those constraints. Therefore, we add an extra objective to our optimization problem. It consists in an additional power variable which, besides the down flex, helps satisfy the energy and power constraints. The power attributed at each instant is optimized with respect to the DA retail price.

Let $\overline{x}, \overline{y}, \underline{x}, \underline{y}, z$ be the decision variables for upward flex, upward regulation, down flex, down regulation, and extra allocated power respectively. The resource allocation problem is formulated as

$$\min_{\substack{\overline{x},\underline{x},\\ \overline{y},\underline{y},z}} \sum_{t=0}^{T_f-1} \left(\sum_{i=1}^{K} \lambda_t^i \overline{x}_t^i \overline{F}_t^i - \sum_{j=1}^{K} \lambda_t^j \overline{y}_t^j \overline{I}_t^j + \sum_{i=1}^{K} \lambda_t^i \underline{x}_t^i |\underline{F}_t^i| - \sum_{j=1}^{K} \lambda_t^j \underline{y}_t^j |\underline{I}_t^j| - \lambda_t^{DA} z_t P_t^{max} \right) \Delta t, \tag{12}$$

where $|.|$ indicates the absolute value, and λ^{DA} is the DA retail price. The first two terms represent the auction for upward flexibility, while the third and fourth terms carry out the down flex auction. The last term determines how much power is allocated at each time step, depending on the DA price. Problem (12) is subjected, for all t, to the following constraints

$$0 \le \overline{x}_t^i, \overline{y}_t^j, \underline{x}_t^i, \underline{y}_t^j, z_t \le 1, \quad \forall i, \forall j, \tag{13a}$$

$$\sum_{i=1}^{K} \overline{x}_t^i \overline{F}_t^i = \sum_{j=1}^{K} \overline{y}_t^j \overline{I}_t^j, \tag{13b}$$

$$\sum_{i=1}^{K} \underline{x}_t^i |\underline{F}_t^i| = \sum_{j=1}^{K} \underline{y}_t^j |\underline{I}_t^j|, \tag{13c}$$

$$z_t P_t^{max} + \sum_{i=1}^{K} \underline{x}_t^i |\underline{F}_t^i| \le P_t^{max}, \tag{13d}$$

$$E_{t+1} = E_t + (z_t P_t^{max} + \sum_{i=1}^{K} \underline{x}_t^i |\underline{F}_t^i|) \Delta t, \tag{13e}$$

$$E_t^{min} \le E_t \le E_t^{max}. \tag{13f}$$

Constraint (13a) limits the clearing volumes to the total available up and down flex, and up and down regulation, at each price point, at time t. It also indicates that the attributed power cannot exceed P_t^{max}. Constraints (13b) and (13c) ensure the equality of supply and demand for the upward and downward market, respectively, at the market clearing point. Constraint (13d) indicates that the total allocated power, comprised of the down flex and the extra power, does not exceed the total maximum power of the device batch. Constraint (13e) corresponds to a simplified model of the evolution of the energy of the entire group of devices, and constraint (13f) ensures that the total energy of the device group remains within an acceptable range.

Problem (12) is convex when the constraints are feasible. The first condition is $t_{run} > t_{max} - t_{run}$. This constraint is satisfied by default as explained in Section 3.2. The second condition is that the power dispatched to each device needs to result in $E_t^i \geq \underline{E}_{t+1}^i$. This condition is guaranteed by Algorithm 2.

5.3. Flexibility and power allocation

The solution of the auction-like problem (12) determines how much flexibility to activate at each price point. Let $\overline{\phi}_t^k$ and $\underline{\phi}_t^k$ be the upward and downward flex, respectively, activated at price k, $1 \leq k \leq K$, at time t. Let also ψ_t be the consumption power decided via optimization. They are computed as

$$\overline{\phi}_t^k = \overline{x}_t^k \overline{F}_t^k, \qquad \underline{\phi}_t^k = \underline{x}_t^k \underline{F}_t^k, \qquad \psi_t = z_t P_t^{max}.$$

To ensure that devices that urgently need to consume are given priority, down flex and consumption power are dispatched first. To do so, the devices asking for the same price are gathered. Within each price bucket, the devices are then ordered by priority. The priority is determined first by the urgency to consume and the remaining time until the device's deadline. A device needs to be allocated power urgently if its energy levels are about to violate the minimum energy requirement, i.e. $E_t^i \leq \underline{E}_{t+1}^i$.

Algorithm 1 Downward flexibility dispatch

1: Input: $\underline{\phi}_t^k$, devices with ask price λ_t^k and $|\underline{f}_t^i| > 0$.
2: Output: f_t^i, flex activated by device i.
3: $\forall i, \lambda_t^i = \lambda_t^k$ rank the devices as
- Devices with $E_t^i < \underline{E}_{t+1}^i$,
- From the earliest to the latest $t_{max}^i - t$.

4: $i \leftarrow 1$.
5: **while** $\underline{\phi}_t^k > 0$ **do**
6: **if** i is a DL **then**
7: $f_t^i \leftarrow \overline{P}^i$.
8: **else**
9: $f_t^i \leftarrow \min(\overline{P}^i, \underline{\phi}_t^k)$.
10: **end if**
11: $\underline{\phi}_t^k \leftarrow \underline{\phi}_t^k - f_t^i$.
12: $i \leftarrow i + 1$.
13: **end while**

Algorithm 1 is repeated for every price point k such that $0 \leq k \leq K$. It indicates that at every price point, devices are selected from the most urgent to the least urgent. All the devices are asked to run at maximum power, except for the last device. Let the superscript I designate the last added device. If the remaining $\underline{\phi}_t^k$ is less than P_t^I, and device I is interruptible, then this device runs on less than P_t^I, namely the remaining $\underline{\phi}_t^k$. In the case where device I is uninterruptible, it has to operate at P_t^I, and the actual activated down flex will exceed the allocated one.

Algorithm 2 describes how the allocated power ψ_t is divided among the devices that did not win the down flex auction. It follows the same pattern as algorithm 1. The only difference is that algorithm 2 guarantees that all the devices that urgently need to consume, i.e. all devices i in the set Ω_{urg} such that $\underline{E}_{t+1}^i < E_t^i$, are dispatched, regardless of whether ψ_t is exceeded or not.

Line 12 of algorithm 2 indicates that not all devices that are allocated power are urgent devices. As the allocated power might be higher than the sum of power urgently needed, other none urgent devices might be dispatched as well. They are ranked following the priority rules defined in algorithm 1.

Algorithm 2 Power dispatch

1: Input: ψ_t, $\Omega_{urg} = \{i \mid \underline{E}_{t+1}^i < E_t^i\}$.
2: Output: P_t^i, consumption of device i.
3: $i \leftarrow 1$.
4: **while** $\psi_t > 0$ and $\Omega_{urg} \neq \emptyset$ **do**
5: **if** i is a DL **then**
6: $P_t^i \leftarrow \overline{P}^i$.
7: **else**
8: $P_t^i \leftarrow \max(\min(\overline{P}^i, \psi_t), (\underline{E}_{t+1}^i - E_t^i)/\Delta t)$.
9: **end if**
10: $\psi_t \leftarrow \psi_t - P_t^i$.
11: **if** $i \in \Omega_{urg}$ **then**
12: $\Omega_{urg} \leftarrow \Omega_{urg} - \{i\}$.
13: **end if**
14: $i \leftarrow i + 1$.
15: **end while**

The remaining devices are allocated upward flexibility, meaning that they will have to curtail their consumption by the quantity computed via algorithm 3. Algorithm 3 is similar in every aspect to algorithm 1. However, due to the absence of urgency for up flex, devices are only ranked according to increasing values $t_{max} - t$, in each price bracket. The reason for this is that the devices with smaller $t_{max} - t$ have a limited window to offer their flexibility, and thus are selected first.

Algorithm 3 Upward flexibility dispatch

1: Input: $\overline{\phi}_t^k$, devices with price λ_t^k and $\overline{f}_t^i > 0$.
2: Output: f_t^i, flex activated by device i.
3: Rank users with same price from earliest $t_{max} - t$ to latest.
4: $i \leftarrow 1$.
5: **while** $\overline{\phi}_t^k > 0$ **do**
6: **if** i is a DL **then**
7: $f_t^i \leftarrow \overline{P}^i$.
8: **else**
9: $f_t^i \leftarrow \min(\overline{P}^i, \overline{\phi}_t^k)$.
10: **end if**
11: $\overline{\phi}_t^k \leftarrow \overline{\phi}_t^k - f_t^i$.
12: $i \leftarrow i + 1$.
13: **end while**

6. Numerical example

In this section, we introduce two scenarios for testing the validity of our approach. The first scenario is a generic flexibility market, where a buyer requests a certain amount of regulation. We will show how this amount is procured in the flexibility market. In the second scenario, we demonstrate how the flexibility market can alleviate instabilities in a low-voltage distribution network where voltage violations arise.

6.1. Use case 1

We have tested our method on a data set containing 200 devices that submit a bid to the platform during the span of one day. The bid contains $\overline{P}$, a price, the type of device (1, 2, 3, for TCLs, EVs, and DL, respectively), t_{min}, t_{max}, and t_{run}. The set contains 4 types of appliances; air-conditioning units considered as TCLs, electric vehicles (EV), washing machines and dishwashers representing deferrable loads.

For simplicity, we have chosen $\Delta t = 1$ h. We also consider the day to range between 00h00 and $T = 24$h00. In this experiment t_{min} and t_{max} were chosen randomly. However, in reality, the HEMS chooses t_{min} and t_{max}, using model-based approaches, or via machine learning. The TP publishes 6 price points for the sellers to choose from. We have chosen

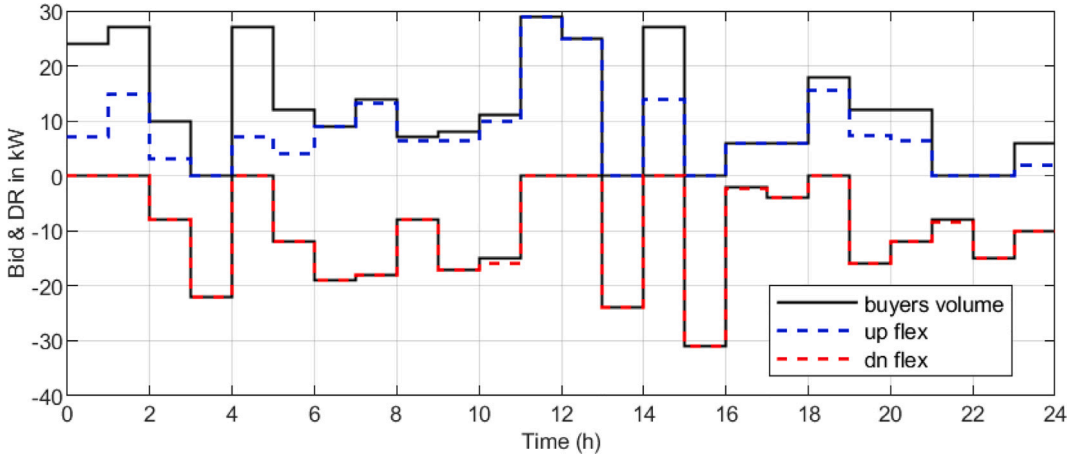


Fig. 4. Auction results: up and down flex vs. buyers' volume.

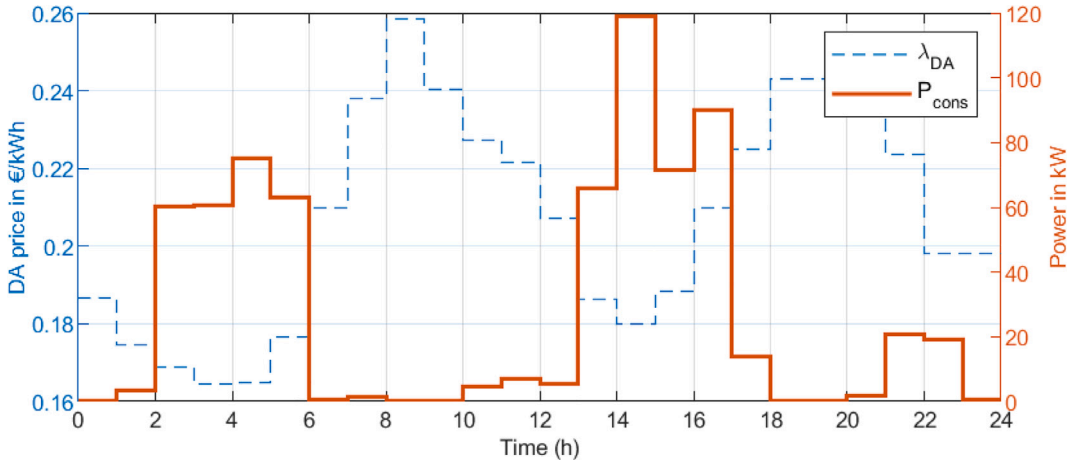


Fig. 5. Sum of power allocated versus DA prices.

price points ranging from $\lambda^1 = 0.1$ €/ kWh to $\lambda^6 = 0.182$ €/ kWh. We assume that there are two buyers who are free to choose their prices.

The sMPC operations are carried out with SCALE-MAMBA [44], a framework that implements actively-secure-with-abort sMPC protocols. This means that if a computing party deviates from the protocol, the honest parties will catch it with overwhelming probability and then abort the procedure. SCALE splits computations into two phases: an offline phase, during which input-independent data is generated, and an online phase, where this data is used to perform the desired computation. Therefore, pre-processed data is generated offline, resulting in faster running times during the online phase. SCALE is based on Shamir Secret Sharing [45], which relies on polynomial interpolation. In this protocol, the secret shares are points over a polynomial whose constant term is the secret value. It is assumed that there is a honest majority among the parties participating in the protocol. Note that sMPC has a high communication overhead, which restricts the efficiency of computations with many computing parties. Thus, in our work, the market participants will not be computing parties themselves. Instead, the sMPC operations will be carried out by three entities, which should have conflicting interests to avoid potential collusion.

Fig. 4 shows the result of the auction for one day. The blue dashed line represents the upward flexibility provided by the consumers, while the red dashed line represents the downward flexibility. The solid black line represents the sum of the initial up and down regulation requested by the buyers. The figure shows that downward flex is always more readily available than up flex. This is due to the fact that device consumption is give priority, so as to respect the device constraints and deadlines. The availability of DR is also tied to the number of available devices. For example, at time t = 00h00 only 8 devices are available, which is not enough to cover the needs of the buyers. DR also depends on the prices asked by the sellers and offered by the buyers, but also on the DA prices. As shown in Fig. 5 when the DA prices are low, device consumption is privileged so as to offer an optimal load schedule for the consumers. A cross examination with Fig. 4 shows that during times of low DA prices, up flex becomes scarce.

Fig. 6 shows that the energy of the available devices always remains within acceptable limits. To check that the limitations of the individual devices are respected, we choose two devices, an EV and a DL (dishwasher), and represent their energy consumption in Fig. 7. The EV shown in red, has deadlines t_{min} = 08h00 and t_{max} = 23h00, and therefore its battery is charged in time, by t = 16h00. The same thing can be said about the dishwasher with t_{min} = 09h00 and t_{max} = 17h00, and which finished its cycle by t = 15h00. Fig. 6 also demonstrates other characteristic of each device type. It can be seen that EV charging is interrupted at t = 11h00 and resumes at t = 13h00. The DL, however, runs continuously until its task is completed.

For the current scenario, the runtime of the online phase of the sMPC steps for the entire auction is around 30 min. All the sMPC parties run identical machines with an Intel i-9900 CPU and 128 GB of RAM. The ping time between the machines is 1.003 ms.

6.2. *Use case* 2

In this use case, we further explore the impact of some of the parameters on the flexibility market. The parameters that we select

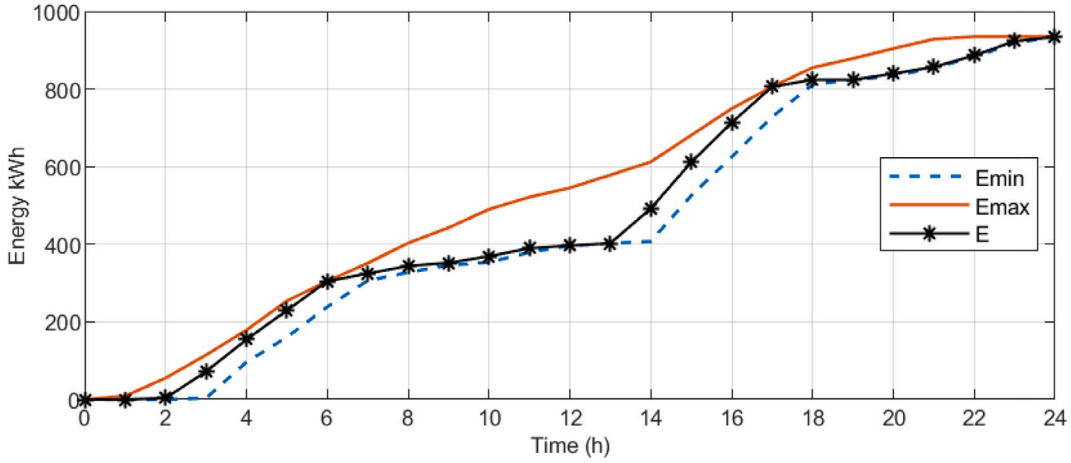


Fig. 6. Sum of energy consumed by all the devices.

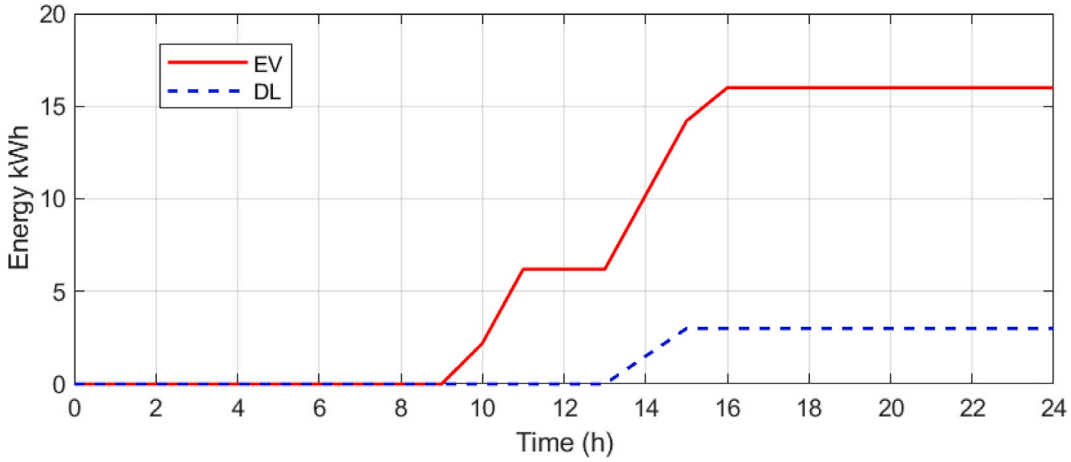


Fig. 7. Sum of energy consumed by an EV and a DL.

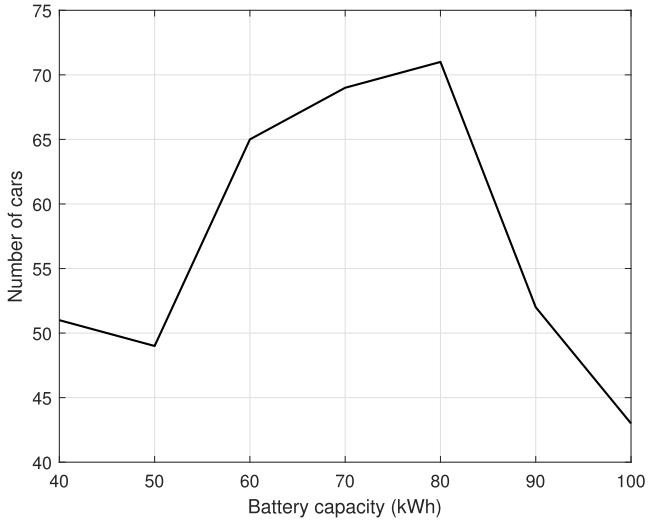


Fig. 8. Battery capacity distribution.

are the price of flexibility selected by the sellers and buyers, and the interval during which a device is available. For the sake of a fair comparison, we restrict our analysis to EVs. The EVs are made to have either random arrival and departure times, or are made to be available for the whole day. The prices, similarly, are chosen to be either random, or are skewed such as the sellers offer the lowest prices, while the buyers select the highest prices. This gives rise to four scenarios that are detailed below. The purpose of this experiment is to guide flexibility providers in selecting their bids, and to determine whether it is more judicious to ask for low prices or to make one's devices available for longer intervals, in order to obtain better returns.

We consider the case of the medium voltage distribution network in [46]. The network is composed of 17 load buses on which 400 EVs are connected, which represents an average of 23 EV per bus. The DSO experiences the imbalance shown on Figs. 10 to 13. It enlists the help of the flexibility market, in which all 400 EVs participate. Even though we consider a homogeneous set of bids all coming from EVs, the EVs themselves exhibit heterogeneous properties. They have varying battery capacities, ranging from 40 kWh to 100 kWh, and have to charge at varying charging speeds. Fig. 8 shows the distribution of battery capacities among the participants. The market is cleared in 15 min time slots, and thus comprises 96 settlement periods ranging from 00h00 to 23h45. The EVs have varying charging rates, and their charging time can vary between 10 and 20 time steps.

6.2.1. *Case random availability, random prices*

In this scenario, the arrival and departure times are selected randomly during the day. Fig. 9 shows the distribution of arrival and departure times for the fleet of EVs. The prices at which the end-users offer their flexibility are also chosen randomly from the following pool,

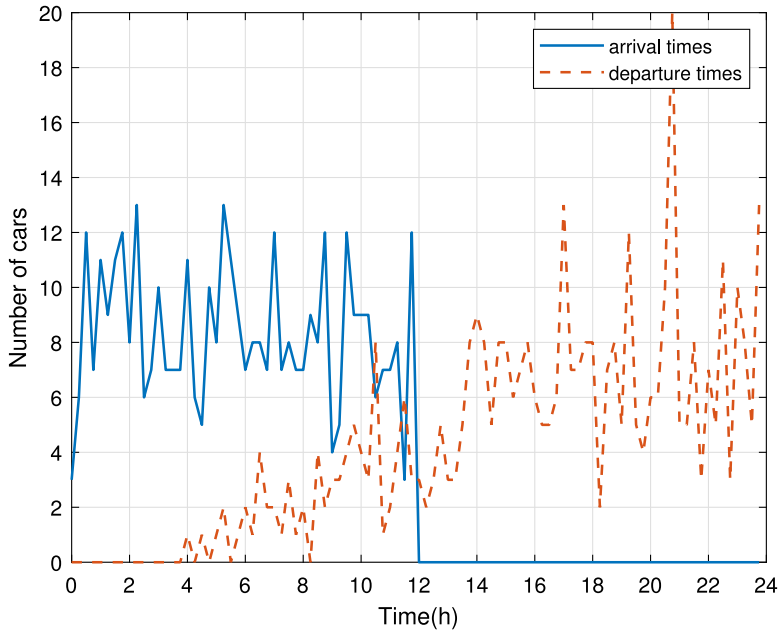


Fig. 9. Arrival and departure times distribution.

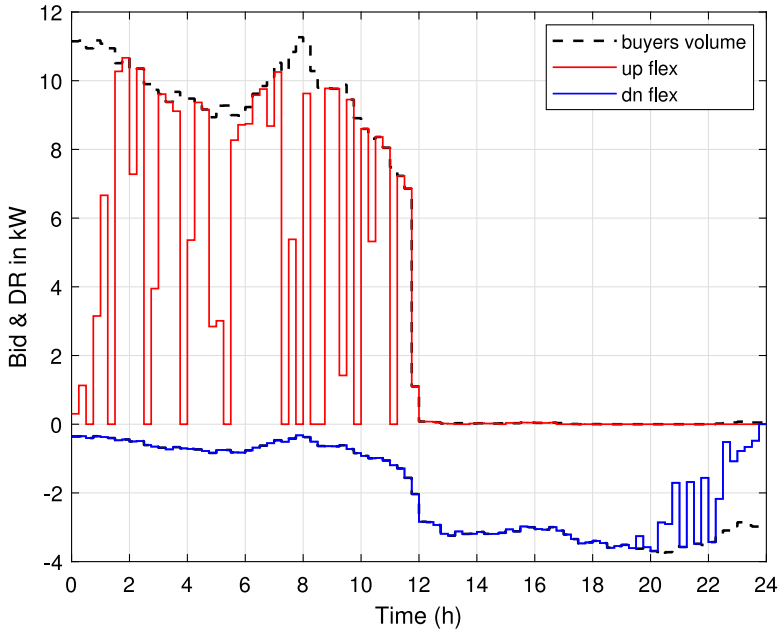


Fig. 10. Market clearance for randomly available EVs with random prices.

from $\lambda = 0.1$ to $\lambda = 0.45$. The DSO, likewise, randomly selects a price from the same pool.

The results of the market clearing as shown on Fig. 10. The dashed black line represents the aggregate upward and downward regulations requested by the DSO. The figure shows that downward flexibility demands were fulfilled to a high percentage whereas upward flexibility is fulfilled to a lesser extent. Computing such percentages gives us 55.7% of the requested upward flexibility found matching bids, versus 85.9% for downward flexibility. A similar phenomenon was observed in the first use case. This could be explained by the fact that downward flexibility brings less discomfort to the users and is therefore easier to provide. We also register the same number of users with accepted bids, 139 for both upward and downward flexibility.

6.2.2. Case random availability, skewed prices

In this scenario, the arrival and departure times are selected randomly, whereas prices are chosen from the diminished pool $\Lambda = \{0.1, 0.15, 0.2\}$, for the sellers. The DSO on the other hand fixed the buying price to 0.45, the highest possible value. Fig. 11 shows the results of market clearing. We notice similar results to the previous case with percentages varying between 50.1% for upward imbalance to 84.9% for downward imbalance. This shows that introducing a bias in prices does not help securing more flexibility for the DSO. This also means that the users could benefit from demanding higher prices for their flexibility. The number of accepted upward and downward market orders has increased, however, to 164 and 141, respectively. The fact that the percentage of fulfilled demand did not increase, means that smaller users, with insufficient power, were selected. Thus, setting

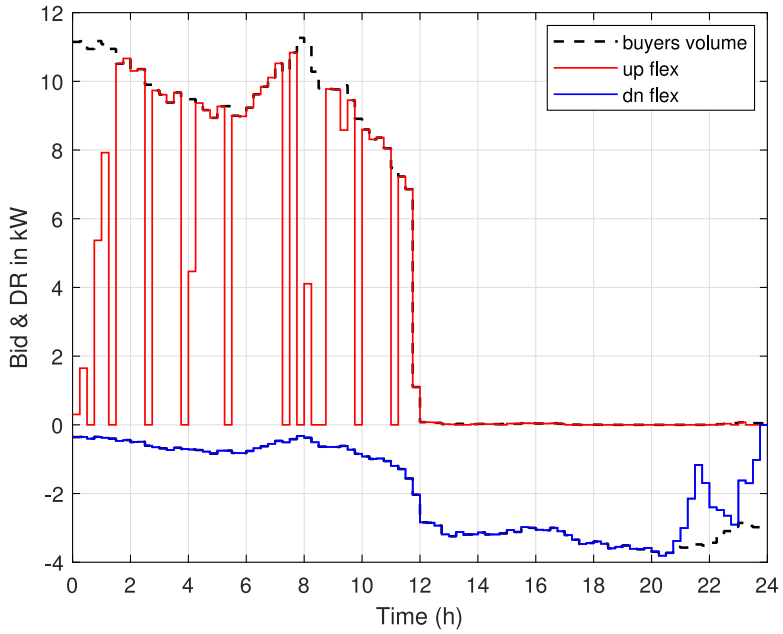


Fig. 11. Market clearance for randomly available EVs with skewed prices.

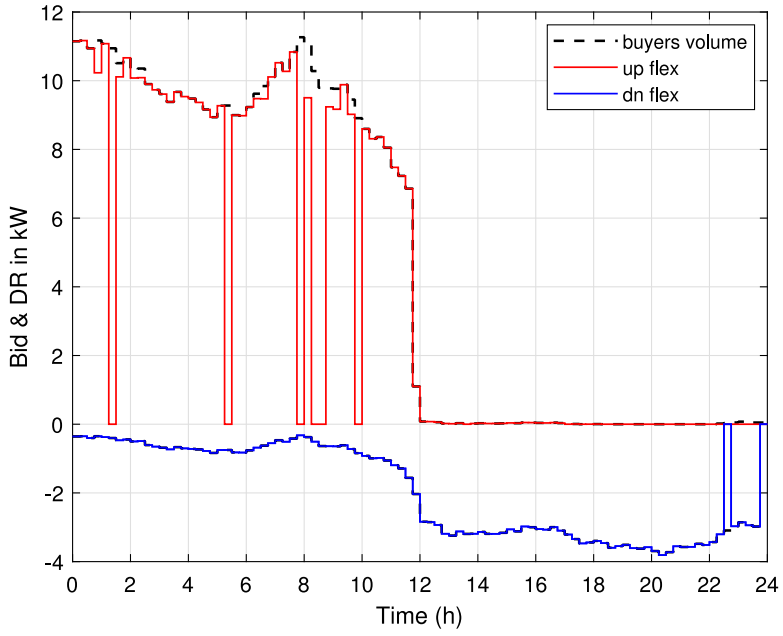


Fig. 12. Market clearance for fully available EVs with random prices.

lower prices could open the path to smaller consumers to take a more active part in DR markets.

6.2.3. Case full availability, random prices

By full availability, we mean that all the EVs set their arrival times as $t_{min} = 1$, and their departure times as $t_{max} = 96$, rendering them available for the whole day. The prices are chosen randomly by both the DSO and the consumers. Fig. 12 contains the results of market clearing. We notice a sharp increase in the amount of fulfilled demand, as attested by the percentages of upward 90.7% and downward 98.4% matched demand. From this, we can start to conclude that the degree of flexibility in arrival and departure times of an EV has more influence on the number of matched bids than the prices. Once again the number of matched downward DR orders, 161, is far superior to the number of matched upward orders, namely 73.

6.2.4. Case full availability, skewed prices

In the last example, the EVs are available for the entire day, while the prices of the sellers are chosen to be low, and the DSO bids the maximum price. Fig. 13 shows the results of market clearing, confirming the trend that longer availability intervals lead to a higher ratio of matched bids. It also confirms that prices have a lesser influence, as the percentage of matched upward demand is 89.7%, and matched downward demand is 98.4%. These numbers are in accordance with the previous case, confirming the importance of flexible intervals over

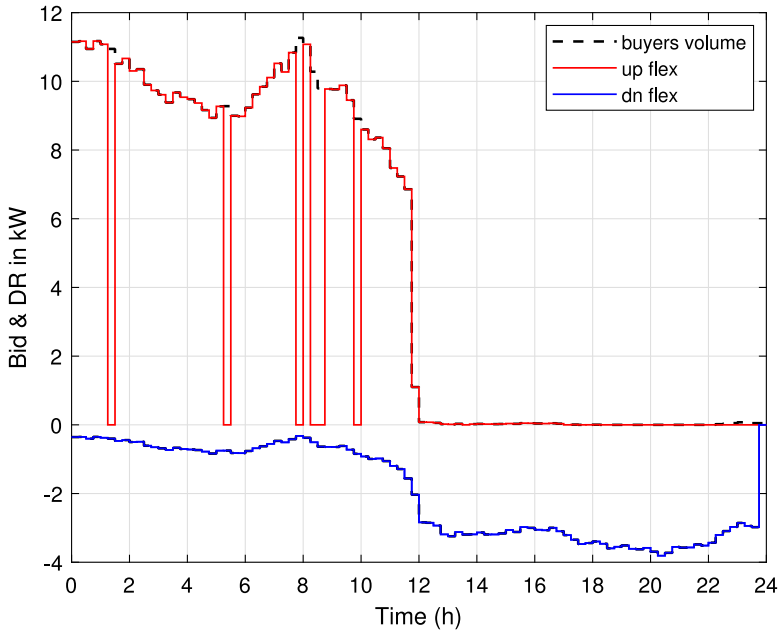


Fig. 13. Market clearance for fully available EVs with skewed prices.

price. The number of EVs that offered upward flexibility is 194, while 123 EVs offered downward flexibility.

7. Conclusion

We presented a privacy-by-design Demand Response market, where buyers and end-consumers can exchange services via a secure trading platform. Both parties can submit their preferences through secretly-shared bids and offers. The market is then cleared without any third-party gaining access to individual consumers' granular data.

A novel method to characterize flexibility is also introduced. It aims at extracting the full flexible potential of each type of device. In this method, the simultaneous upward/downward flexibility potential of each device, as well as its power and energy constraints are linked to the type of device, user comfort intervals, power ratings, and prior power allocation. The computations also simple enough to be carried out via sMPC. In subsequent steps, aggregation takes over to ensure the privacy of the end-consumers, and to clear the market.

In some instances, revealing aggregated data might represent a risk of information leakage. Thus, in future works, we would like to also use sMPC to solve the linear programming problem for social welfare maximization. The use of sMPC for solving linear programs using the simplex method was already proposed in [47]. In order to solve the social welfare maximization problem, we would need to perform further benchmarking and possibly improve the existing sMPC protocol for the simplex method in order to ensure the efficiency and scalability of the resulting algorithm.

Another area that remains to be explored is the effect of the flexibility market on the medium/low voltage distribution grid. In such analysis, we explore additional conditions for the flexibility market to relieve the stress on the grid. These conditions include the minimum number of participants, the types of devices that have the largest impact, and the maximum number of deviations allowed before the grid is negatively impacted.

CRediT authorship contribution statement

Fairouz Zobiri: Conceptualization, Methodology, Software, Writing – original draft. **Mariana Gama:** Software, Writing – original draft. **Svetla Nikova:** Supervision, Writing – review & editing. **Geert Deconinck:** Supervision, Writing – review & editing.

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

Data will be made available on request.

Acknowledgments

This work was supported by the Research Council KU-Leuven: C16/15/058, by the Research Foundation - Flanders through FWO SBO project SNIPPET S007619, by CyberSecurity Research Flanders with reference number VR20192203, and by ERC Advanced Grant ERC-2015-AdG-IMPaCT.

References

[1] European Environment Agency. New registrations of electric vehicles in Europe. 2021, https://www.eea.europa.eu/ims/new-registrations-of-electric-vehicles. Accessed: 2022-03.
[2] Du P, Lu N, Zhong H. Demand response in smart grids. Springer Cham; 2019.
[3] Rahimi F, Ipakchi A. Demand response as a market resource under the smart grid paradigm. IEEE Trans Smart Grid 2010;1(1):82–8.
[4] Chao H. Demand response in wholesale electricity markets: The choice of customer baseline. J Regul Econom 2011;39(1):68–88.
[5] Afzalan M, Jazizadeh F. Residential loads flexibility potential for demand response using energy consumption patterns and user segments. Appl Energy 2019;254:113693.
[6] Hu J, Yang G, Ziras C, Kok K. Aggregator operation in the balancing market through network-constrained transactive energy. IEEE Trans Power Syst 2019;34(5):4071–80.
[7] Ur-Rehman O, Zivic N, Ruland C. Security issues in smart metering systems. In: 2015 IEEE international conference on smart energy grid engineering (SEGE). 2015, p. 1–7. http://dx.doi.org/10.1109/SEGE.2015.7324615.
[8] Lu S, Samaan N, Diao R, Elizondo M, Jin C, Mayhorn E, Zhang Y, Kirkham H. Centralized and decentralized control for demand response. In: ISGT 2011. 2011, p. 1–8.
[9] Sakurama K, Miura M. Communication-based decentralized demand response for smart microgrids. IEEE Trans Ind Electron 2017;64(6):5192–202.
[10] Wang Y, Wu L, Wang S. A fully-decentralized consensus-based ADMM approach for DC-OPF with demand response. IEEE Trans Smart Grid 2017;8(6):2637–47.

[11] Bahrami S, Amini MH, Shafie-khah M, Catalão JPS. A decentralized electricity market scheme enabling demand response deployment. IEEE Trans Power Syst 2018;33(4):4218–27.
[12] Dusparic I, Taylor A, Marinescu A, Cahill V, Clarke S. Maximizing renewable energy use with decentralized residential demand response. In: 2015 IEEE first international smart cities conference (ISC2). 2015, p. 1–6. http://dx.doi.org/10.1109/ISC2.2015.7366212.
[13] Yang X, Wang G, He H, Lu J, Zhang Y. Automated demand response framework in ELNs: Decentralized scheduling and smart contract. IEEE Trans Syst Man Cybern Syst 2020;50(1):58–72.
[14] Aggarwal S, Kumar N. A consortium blockchain-based energy trading for demand response management in vehicle-to-grid. IEEE Trans Veh Technol 2021;70(9):9480–94.
[15] Pop C, Cioara T, Antal M, Anghel I, Salomie I, Bertoncini M. Blockchain based decentralized management of demand response programs in smart energy grids. Sensors 2018;18(1).
[16] Wen S, Xiong W, Tan J, Chen S, Li Q. Blockchain enhanced price incentive demand response for building user energy network in sustainable society. Sustainable Cities Soc 2021;68:102748.
[17] Park K, Lee J, Das AK, Park Y. BPPS:Blockchain-enabled privacy-preserving scheme for demand-response management in smart grid environments. IEEE Trans Dependable Secure Comput 2023;20(2):1719–29.
[18] Jindal A, Aujla GS, Kumar N, Villari M. GUARDIAN: Blockchain-based secure demand response management in smart grid system. IEEE Trans Serv Comput 2020;13(4):613–24.
[19] Gong Y, Cai Y, Guo Y, Fang Y. A privacy-preserving scheme for incentive-based demand response in the smart grid. IEEE Trans Smart Grid 2016;7(3):1304–13.
[20] Xue K, Yang Q, Li S, Wei DS, Peng M, Memon I, Hong P. PPSO: A privacy-preserving service outsourcing scheme for real-time pricing demand response in smart grid. IEEE Internet Things J 2019;6(2):2486–96.
[21] Parizy ES, Bahrami HR, Choi S. A low complexity and secure demand response technique for peak load reduction. IEEE Trans Smart Grid 2019;10(3):3259–68.
[22] Rahman MS, Basu A, Kiyomoto S, Bhuiyan MZ. Privacy-friendly secure bidding for smart grid demand-response. Inform Sci 2017;379:229–40.
[23] Li H, Lin X, Yang H, Liang X, Lu R, Shen X. EPPDR: An efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid. IEEE Trans Parallel Distrib Syst 2014;25(8):2053–64.
[24] Li D, Yang Q, Yu W, An D, Yang X, Zhao W. A strategy-proof privacy-preserving double auction mechanism for electrical vehicles demand response in microgrids. In: 2017 IEEE 36th international performance computing and communications conference (IPCCC). 2017, p. 1–8. http://dx.doi.org/10.1109/PCCC.2017.8280481.
[25] Kumari A, Tanwar S. A reinforcement-learning-based secure demand response scheme for smart grid system. IEEE Internet Things J 2022;9(3):2180–91.
[26] Sarenche R, Salmasizadeh M, Ameri MH, Aref MR. A secure and privacy-preserving protocol for holding double auctions in smart grid. Inform Sci 2021;557:108–29.
[27] Shamir A. Identity-based cryptosystems and signature schemes. In: Advances in cryptology. Springer Berlin Heidelberg; 1985, p. 47–53.
[28] Goldwasser S, Micali S, Rackoff C. The knowledge complexity of interactive proof systems. SIAM J Comput 1989;18(1):186–208. http://dx.doi.org/10.1137/0218012.
[29] Pedersen TP. Non-interactive and information-theoretic secure verifiable secret sharing. In: Advances in cryptology — CRYPTO '91. Springer Berlin Heidelberg; 1992, p. 129–40.
[30] McParland C. OpenADR open source toolkit: Developing open source software for the smart grid. In: 2011 IEEE power and energy society general meeting. 2011, p. 1–7. http://dx.doi.org/10.1109/PES.2011.6039816.
[31] Elgamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans Inform Theory 1985;31(4):469–72. http://dx.doi.org/10.1109/TIT.1985.1057074.
[32] Schnorr CP. Efficient identification and signatures for smart cards. In: Brassard G, editor. Advances in cryptology — CRYPTO' 89 proceedings. New York, NY: Springer New York; 1990, p. 239–52.
[33] Zobiri F, Gama M, Nikova S, Deconinck G. A privacy-preserving demand response market based on multi-party computation. In: 2022 IEEE power & energy society innovative smart grid technologies conference (ISGT). 2022, p. 1–5.
[34] Vandael S, Claessens B, Hommelberg M, Holvoet T, Deconinck G. A scalable three-step approach for demand side management of plug-in hybrid vehicles. IEEE Trans Smart Grid 2013;4(2):720–8.
[35] Smart NP. Cryptography made simple. Springer; 2016.
[36] Bogetoft P, Christensen DL, Damgård I, Geisler M, Jakobsen T, Krø igaard M, Nielsen JD, Nielsen JB, Nielsen K, Pagter J, Schwartzbach M, Toft T. Secure multiparty computation goes live. In: Financial cryptography and data security. Springer Berlin Heidelberg; 2009, p. 325–43.
[37] Mustafa MA, Cleemput S, Aly A, Abidin A. An MPC-based protocol for secure and privacy-preserving smart metering. In: 2017 IEEE PES innovative smart grid technologies conference Europe (ISGT-Europe). 2017, p. 1–6. http://dx.doi.org/10.1109/ISGTEurope.2017.8260202.
[38] Abidin A, Aly A, Cleemput S, Mustafa MA. An MPC-based privacy-preserving protocol for a local electricity trading market. In: Foresti S, Persiano G, editors. Cryptology and network security. Cham: Springer International Publishing; 2016, p. 615–25.
[39] Mustafa MA, Cleemput S, Aly A, Abidin A. A secure and privacy-preserving protocol for smart metering operational data collection. IEEE Trans Smart Grid 2019;10(6):6481–90. http://dx.doi.org/10.1109/TSG.2019.2906016.
[40] Cramer R, Damgård IB, Nielsen JB. Secure multiparty computation and secret sharing. Cambridge University Press; 2015.
[41] Sadeghianpourhamami N, Deleu J, Develder C. Definition and evaluation of model-free coordination of electrical vehicle charging with reinforcement learning. IEEE Trans Smart Grid 2020;11(1):203–14.
[42] Wang Y, Zobiri F, Deconinck G. Prosumer centric flexibility market with geometric Brownian motion pricing method. In: 2021 IEEE PES innovative smart grid technologies Europe (ISGT Europe). 2021, p. 1–6. http://dx.doi.org/10.1109/ISGTEurope52324.2021.9639936.
[43] Petersen MK, Edlund K, Hansen LH, Bendtsen J, Stoustrup J. A taxonomy for modeling flexibility and a computationally efficient algorithm for dispatch in smart grids. In: 2013 American control conference. 2013, p. 1150–6. http://dx.doi.org/10.1109/ACC.2013.6579991.
[44] Aly A, Keller M, Orsini E, Rotaru D, Scholl P, Smart NP, Wood T. SCALE and MAMBA documentation. 2018.
[45] Shamir A. How to share a secret. Commun ACM 1979;22(11):612–3.
[46] Grady W, Samotyj M, Noyola A. The application of network objective functions for actively minimizing the impact of voltage harmonics in power systems. IEEE Trans Power Deliv 1992;7(3):1379–86.
[47] Toft T. Solving linear programs using multiparty computation. In: Dingledine R, Golle P, editors. Financial cryptography and data security. FC 2009. Lecture notes in computer science, vol. 5628, Berlin, Heidelberg: Springer; 2009.