

Blockchain Based Decentralized Framework for Energy Demand Response Marketplace

Varun Deshpande
LIGM, UPE, ESIEE Paris
Noisy-le-Grand, France
varun.deshpande@esiee.fr

Laurent George
LIGM, ESIEE Paris
Noisy-le-Grand, France
laurent.george@esiee.fr

Hakim Badis
LIGM, UPEM
Champs-sur-Marne, France
hakim.badis@u-pem.fr

Alemayehu Addisu Desta
LIGM, ESIEE Paris
Noisy-le-Grand, France
alemayehu.desta@esiee.fr

Abstract—In the context of smart grids, Demand Response (DR) is used to manage energy imbalance by smoothing consumption peaks through voluntary rationing of energy by participants. However, it largely remains centralized and opaque with little to no traceability. To resolve this, we propose a blockchain-based framework in which a consortium of DR allotters and certifying authorities maintain the blockchain. This brings in more transparency, traceability, and complete decentralization along with trustlessness, non-repudiation, and immutability. Further, the framework uses distinct components/concepts like Secure Elements, Escrow Accounts, Applied Smart Contracts in unison to fix the impediments of previous blockchain-based propositions. Next, we propose a fair and efficient DR allotment mechanism for a distributed DR marketplace whose execution time is less than 1 minute for more than 20,000 participants. Further, through simulations, we show the impact of different parameters on it and demonstrate its ability to delicately balance various paradigms of DR metrics. Finally, we conclude with our findings on system reliability and its inordinate effects on DR allotment metrics.

Index Terms—blockchain, demand response, energy, smart contracts, escrow accounts, secure element, allotment model.

I. INTRODUCTION

Blockchain, as a concept, has shown tremendous development and subsequently evolved itself during the last decade. Although the concept was introduced in 1990 [1], the breakthrough came when the first commercial prototype, i.e., the Bitcoin cryptocurrency [2] emerged in the market. Since then, the economics of blockchain ecosystem are scaling up. Blockchain can be used for a variety of different applications where trustlessness and decentralization are indispensable. This has led to adoption of blockchain across a range of domains such as IoT [3], Big data [4], Supply chain [5], etc.

In the energy domain, blockchain has been used to effectively solve the problem of peer to peer (P2P) energy trading in micro-grids [6]. Because of the immutability and traceability offered by blockchain, the transaction (tx) trail of the P2P energy trading (from a verified origin) becomes authentic even in a trustless environment. A similar concept can be applied to demand response (DR) in the energy domain where needs for transparency and payment guarantee are high.

DR refers to the change in power consumption pattern of the consumers to match the power demand to the power supply in return for incentives (mostly economic) and cost reduction. The concept of DR is size agnostic and hence can be applied

to localized as well as macro grids. The DR for localized grids is managed by Distribution System Operator (DSO) while for macro-grids is managed by Transmission System Operator (TSO). In this work, we consider TSO-managed DR for macro-grids in a DR marketplace.

While the technology needed for communication and monitoring in DR is already there, the challenges for optimization and control are not yet solved effectively [7]. As more and more end consumers are participating in the DR sessions, the fair DR execution is becoming more vital. Further, the issues of guaranteed payments after DR action needs to be addressed. The ability of blockchain to work in distributed and trustless environment can be well utilized to solve these core DR problems bringing in more trust, transparency, and immutability. In this paper, we propose a blockchain-based framework to solve these issues. The novelty of our proposed framework is that it proposes complete decentralization of all the activities taking place in a typical DR session and can work in a trustless and distributed environment using its blockchain backbone and also meticulously dodges blockchain's scaling issue for big data storage.

The remainder of this paper is organized as follows: In Sec. II, we provide a literature review on related works. Sec. III describes the components and concepts used in our framework. Sec. IV explains the DR allotment model used by our framework. In Sec. V, we describe the working principle of the framework using the components/concepts already explained in Sec. III. Sec. VI explain the results of the simulations. Finally, we conclude our paper in Sec. VII with a brief overview of future work.

II. RELATED WORK

There has been increasing interest in the field of DR and its implementation based on blockchain. In [8], the authors discussed DR in the context of smart grids. When the bid for DR is verified by the DSO, blockchain is used as a back-end to store the subsequent smart meter values for validation of DR actions and to serve as a proof for calculating rewards using Smart Contracts (SCs) which are implemented on the same blockchain. In [9], the authors improve the energy management using game theory approach in a distributed smart grid. Blockchain is used as a Service (BaaS) implementing P2P energy trading. P2P energy trading is also discussed in [10] with

a special focus on privacy. The authors addressed the privacy issue by using a consortium blockchain (more appropriately called permissioned blockchain¹). Wu *et al.* [11] proposed a machine-to-machine blockchain based on Multichain [12] framework to manage DR within a smart grid. They discussed the concept of automatic payments using SCs but they did not implement it in their presented Proof of Concept.

In [13], the authors implemented a blockchain-based PoC on an industrial operating system (Predix platform) where the energy production and consumption are managed using the concept called “green certificates”, i.e., extra power generation at one site is certified by issuing certificates which are then sold to satisfy any shortfall in energy at another site. The blockchain supports multi-assets for better management.

To summarize, knowing that a given DR session is divided into three parts, occurring in order viz., allotment (rationing of the DR signal among the interested participants), monitoring (observing the changed pattern of power consumption) and settlement (payment for the voluntary change of power consumption), most of the previous works in this domain focused mainly on decentralizing the DR monitoring and DR settlement part using blockchain.

To the best of our knowledge, no other work has proposed decentralization of DR allotment part. Our work tries to fill this exact niche, i.e., decentralizing the DR allotment through a decentralized bidding process implemented using blockchain and SC. This is important because DR allotment is the first and most important part in a DR session as it decides who are the clients participating in the DR session and also how much they are going to be paid. In all the previous works, this part was centralized making the whole DR session only partially decentralized. This is not an interesting preposition in real world’s implementation perspective because still single entity controls the most critical step in a given DR session. Through clever use of blockchain technology, our proposed framework overcomes this bottleneck.

Further, for DR monitoring, we make use of smart meters that are equipped with Secure Elements (SEs). We explain how they provide secure, tamper-proof power consumption values and also why it is not necessary to store this data on the blockchain. Next, we use escrow accounts based on blockchain for DR settlement and show how they are better suited for payments compared to SCs.

III. DR FRAMEWORK AND ITS COMPONENTS

A typical DR session involves a wide range of heterogeneous systems coordinating together. Our proposed framework illustrated in Fig. 1 likewise uses a wide range of components and concepts which are described in detail in this Sec. through which it achieves complete decentralization of the DR session viz., allotment, monitoring, settlement (explained in Sec. V). In our framework, TSOs and market regulators (tier 1) maintain a blockchain which serves as a backbone database for the DR participants (tier 2). The participating bids for DR are collected by all the tier 1 participants and allotment is collectively

¹access to the blockchain is granted after permission for the participants.

decided by them using consensus. To have a synchronized gate open/close timing, each tier 1 participant is equipped with a GPS clock. For *proof of origin*, tier 2 participants are equipped with an SE.

A. Blockchain

Blockchain can be defined as a chain of data blocks linked together sequentially where each block is chained using the hash of its previous block [14]. Blockchain classification is explained in [15]. In our framework, we propose to use a private permissioned blockchain because, in the case of data transactions, the pre-requisite access permissions are imperative to guarantee trusted results and privacy.

Next, blockchain platforms, in general, are optimised for a variety of use-cases based on different end-user needs. For e.g., some platforms (Hyperledger Fabric [16], Multichain [12], etc.) allows efficient data/finance transaction management while some platforms (Ethereum [17], Hyperledger Burrow [18], etc.) are optimized for running code. In our work, we propose to use blockchain platforms optimised for efficient transaction management as there are more data/finance transactions to manage in a given DR session compared to only one code running requirement (DR allotment model) [19] (see Sec. III-B). In our framework, blockchain serves as an all-in-

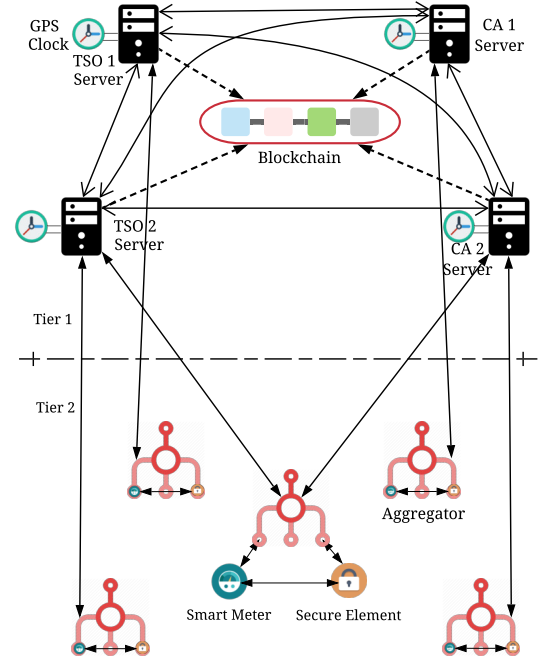


Fig. 1. Blockchain-based framework for DR marketplace management.

one backbone. There are two tiers of participants, i.e., tier 1 participants maintain the blockchain and tier 2 participants access the blockchain as BaaS. With this two-tier system, the issue of security vulnerability does not arise because the blockchain is maintained by tier 1 participants only (i.e. blockchain maintenance is not split across tiers). Next, this two-tier system helps to increase trustworthiness further as all the participants in tier 1 are trusted and only admitted after high scrutiny.

Tier 1 consists of TSOs and certifying authorities (CAs) who form a fully connected P2P network over a secure and reliable Virtual Private Network. CAs in tier 1 can either be DR marketplace regulators run by governments or delegated organizations acting as independent watchdogs to monitor DR marketplaces. Although the blockchain is accessible to all the participants of the DR marketplace, the writing rights (permission to form new blocks) is accorded only to tier 1 participants. Tier 2 consists of independent aggregators in the DR session who access blockchain as BaaS through tier 1 participants. During a DR session, participating aggregators (after monitoring the consumption pattern) pass their respective *calculated* bids to one or more tier 1 participants. However, to be resilient against tier 1 node failures, we insist that they always pass their bids to more than one tier 1 participants. Further, tier 1 participants maintain a synchronized tx pool. All the bids received individually by tier 1 participants are pushed in their respective individual tx pool. As all these tx pools are synchronized, the changes appear in tx pools of other tier 1 participants as well.

The tx pools synchronization can be done using Reliable Multicast [20]. Many different protocols are implementing Reliable Multicast [21]. A Reliable Multicast enforces a total order of the updates preserving the coherency of the distributed information. The rationale behind the proposition of common synchronized tx pool in our framework is to ensure coherent output during DR allotment as the selection model is applied by each tier 1 participant in a distributed setting on the same set of bids from the synchronized tx pool.

Further, a model to be used for selecting the appropriate subset of aggregators during the DR session is first proposed by TSO. The model is then subsequently verified by the CAs in tier 1 through consensus and then added to the blockchain. Subsequent DR sessions then can use the newly certified model. The certification of the models helps to add an extra layer of security. For example, the certified model can favor aggregators based on given parameters (like reliability, consistency, punctuality, etc.) but the verification and certification of the models make sure that they are not biased or partial against invalid criteria (like favoring aggregators affiliated to political parties, etc.). As CAs collectively validate the model (through consensus), it increases the transparency between the aggregators and TSOs. Adding the certified selection model on the blockchain subsequently makes it immutable thereby securing future DR sessions.

The important property of blockchain which makes it indispensable in our framework is that it allows the DR participants to apply consensus for different decisions. The property of collective decisions is very unique to a blockchain. In the context of DR allotment, this collective decision property through consensus is of prime importance. This is because if a single entity (TSO) is the sole decision-maker, the decentralization of subsequent steps makes little sense. Even if decentralization solves trust issues for subsequent steps (e.g. payment), the prospect of not being allotted any DR quota at first place outweighs everything.

Further, consensus decisions in blockchain are inherently irreversible because of the collective nature of the decisions. Connecting this with the linking structure of blockchain, any data-editing possibility is ruled out till $> 50\%$ network deviates from its intended behavior. This guarantees immutability and a complete trace of past DR sessions. Next, features like native support for asset handling, multi-signature wallets (see Sec. III-C) makes it apt in the context of DR.

B. Applied Smart Contracts

SCs are a type of computer code stored on a blockchain, implementing a given logic. To some extent, one can draw an analogy between them and classes in programming language's literature. SCs, in technical terms, do not run on the blockchain. As blockchain is just a way to store data across all the participating nodes, it is used to distribute SCs across all the participating nodes. Because we are using a blockchain optimized for transaction management (refer Sec. III-A), it is necessary to implement the concept of smart contracts separately for our proposed blockchain framework. To distinguish it from existent smart contract implementation, we refer our proposed implementation as Applied Smart Contract or ASC.

The process for ASC implementations starts by codifying a given logic. The given logic in our case is the DR allotment model. The new DR allotment model is first codified, compiled and then converted to byte code (to form a single binary) by the concerned TSO. This binary (ASC) is then subsequently encapsulated within a standard blockchain tx. This is then added to the synchronized tx pool by the TSO. CAs from tier 1 then certifies it by evaluating it on various criteria like fairness, time of execution, etc. Once certification is finished by CAs individually, the certified DR allotment model is then published on the blockchain (through consensus) and thereby subsequently stored across all the participating nodes.

During the DR allotment, when this ASC has to be executed, all tier 1 participants retrieve it and run it and verify the allotment in a distributed setting. At the end of the execution, a new block containing the output of the DR model (list of selected and rejected bids) is formed and added to the blockchain through consensus (similar to the process of adding a new DR allotment model on the blockchain). Each node verifies this block before adding it to its local blockchain copy. The consensus/verification is achieved knowing the fact that the same DR allotment model, replicated across all the tier 1 nodes, run with an identical interpreter, applied on the same data-set (common synchronized tx pool), will yield the same output. We propose to use distributed consensus between identified block validators [12] which works like practical Byzantine Fault Tolerance but with a caveat i.e. single validator per block. Detailed execution steps of consensus using an example simulation are discussed in Sec. VI.

Even with the great similarity in the base concept, the proposed ASC differs from SC in many aspects. Starting with the language support, ASC can be in any language while SC has to be in the language supported by the blockchain

platform. Because of this flexibility, ASC does not need an installation/maintenance of separate Virtual Machine (VM) unlike in the case of SC where running a separate VM is indispensable. For example, in the case of Ethereum, running Ethereum VM is absolutely necessary. Next, running a separate VM has a large overhead and sharing the live state of variables within the SC with participating nodes (using underlying P2P communication protocol) further increases it. As ASC doesn't need VM per se, the overhead for running the blockchain is significantly lower.

However, as VM is not necessary in the case of ASC, maintaining the live state of a variable in the code/ASC has to be managed separately using proper indexing/retrieval mechanism if the need arises. To summarize, our proposed ASC offers most of the common features of the SC, i.e., immutable-persistent storage of data/code, trace of the txs, trace of ownership of data, etc., with much-reduced overhead because of the absence of a dedicated VM.

C. Escrow Accounts

In our framework, for DR settlement, we propose to use a decentralized escrow account setting. An Escrow account, in its simplest form, is an account where the asset is held in the interim till the transaction between the two parties is completed [22]. The escrow account is directly controlled by an external third party which is not involved in the transaction under the hypothesis that it is unbiased, impartial and trustworthy. When escrow account setting is implemented on the blockchain, the trusted third party is effectively replaced by the blockchain itself thereby making the escrow setting truly decentralized and unbiased. Escrow account setting on a blockchain can be implemented in two ways i.e., 1) Using SCs (like LocalEthereum [23] implemented on Ethereum Blockchain), 2) Through native support provided by specific blockchain platforms (like multi-signature wallets implemented in Multichain platform [12]).

In our framework, we propose to use the escrow account setting implemented using “multi-signature wallet on the blockchain” as it is faster, lighter and simpler due to: 1) absence of an extra VM, 2) Native support on the blockchain platform. Also, in the case of SCs, the code needs absolute certification, as improper SCs implementation may lead to payment vulnerabilities (like the DAO attack [24]). In a decentralized escrow account based on the multi-signature wallet, the inflow of funds/tokens is unrestricted but the outflow is restricted by a constraint of multiple signatures. For instance, if a decentralized escrow account is owned by 3 entities on a blockchain, to make any outflow payment, 2 entities (majority) need to sign the same outflow transaction to validate the payment execution.

Further, an escrow setting based on multi-signature wallets does not have scaling, timing, and overhead issues because the payment decisions are done off-chain by the concerned entities and only the payment execution is carried out by each participant (involved in the tx) on the blockchain. Further, it is truly decentralized as all participating entities in the tx have

an equal say. Fig. 2 explains the typical DR settlement flow. In our framework, for implementing an escrow setting, we propose to use 3-signature wallets i.e., wallets jointly owned by three entities. For each participating aggregator, there is a 3 signature wallet co-owned by that specific aggregator², the concerned TSO and the third party/CA. To withdraw the assets from these accounts, any 2 out of 3 signatures are required (TSO-Aggregator or TSO-CA or Aggregator-CA). The assumed hypothesis for the third party/CA is that it is trusted and unbiased. The problems arising in case of its misbehavior are out of the scope of the paper. After the DR session starts,

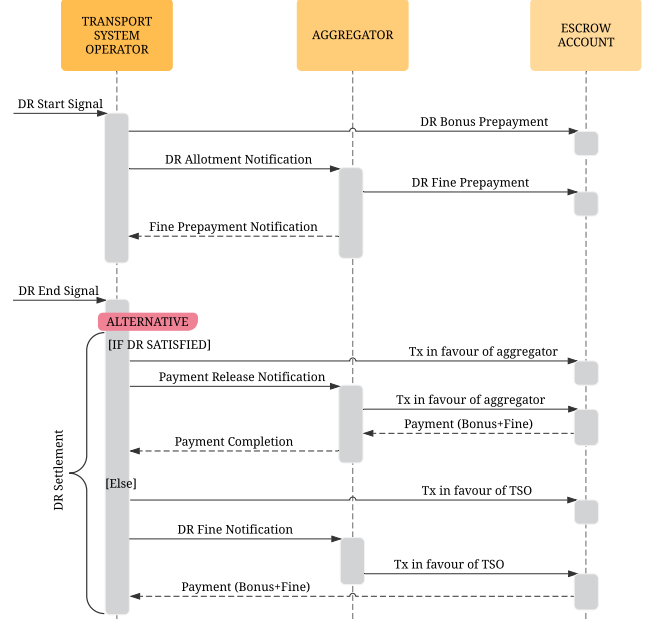


Fig. 2. A typical DR settlement flow using the escrow account.

the DR preposition bids are collected by tier 1 from the tier 2 participants and the DR allotment model is run in a distributed setting. Once the subset of aggregators is selected, for each aggregator, the TSO deposits the bonus in advance in the respective escrow accounts corresponding to each aggregator. After the advance payment, the aggregator(s) are asked to pay an advance fine in the same corresponding account in case if the aggregator(s) does not respect the agreed DR power consumption reduction. After the end of DR monitoring, a tx containing the payment and the fine is then signed by TSO in favor of the aggregator if the allotted power reduction baseline is followed else in its favor. This tx is then forwarded to the corresponding aggregator for signature after which the payment is done from the escrow account to the pertinent account. In case of discrepancy, either of the entities won't sign it and the decision then rests with the third party/CA which would do arbitration based on the data from the DR monitoring session.

D. Secure Element

An SE [25] is a small (25 mm²), secure by design, programmable micro-controller which provides trusted execution

²separate wallets facilitates the easy financial auditing.

and storage environment (TEE/TSE). It comes with an integrated crypto-processor and is specially designed to implement cryptographic functions (encryption-decryption, hash, signature, and its verification, etc.) in a fast and tamper-proof way. SE is used on a large-scale in various fields like payment (chip-based credit/debit cards), communication (SIM cards), identification documents (biometric passports, chip-based ID cards), IoT [26] (for identification, authentication, certification, verification), etc. It is also used to establish *root of trust* in Blockchain-IoT systems [27].

In our framework, SE is used in signing the DR bids and in DR monitoring. It is used as a Hardware Security Module (HSM) add-on by the aggregators (see Fig. 1). The SE stores the private key (used for signing the data) in its TSE. TSE ensures not only that the private key cannot be leaked/copied but also the authenticity of data in DR bids and DR monitoring. The corresponding public key is bound to the identity of the aggregator having that particular SE using Public Key Infrastructure (PKI) [28] and then shared with the concerned parties (TSO, CAs, etc.). The decentralization of the DR session does not affect it as: 1) The process of identity binding is carried out for all new aggregators before they can start participating in any DR session, 2) The identity binding is carried out by appropriate registration authority and is point-to-point. Further discussion on the process of identity binding using PKI is out of the scope of the paper.

When the DR session starts, aggregators decide on the amount of power consumption reduction to bid. This bid is then sent to SE for signature which is then ultimately transferred to Blockchain using BaaS through tier 1 participants. As the private key is stored in the SE, the tier 1 participants can have the surety that the bid indeed came from an authorized aggregator and is not changed during transmission³.

For DR monitoring, we propose the use of a smart meter. This smart meter is connected to the SE of the aggregator (see Fig. 1). When the smart meter records data (power consumption value), it is sent to the SE for signature. Once the data is signed, the authenticity of the data can then easily be trusted. This can be reasoned further by understanding the hardware integrity of the smart meter. The smart meter's hardware integrity is managed by the SE's tamper interrupt pin. Hence, when the smart meter is tampered, the SE detects it through its tamper interrupt pin and subsequently stops signing the data. Considering this, when the signed data arrives from this smart meter, one could guarantee that the smart meter is not tampered. Further, on verifying the signature, the receiving entity can be sure that the data they received is of verified origin and unchanged during transmission (else signature verification would fail).

Furthermore, we propose that during the DR session, this signed power consumption data is stored by the aggregator (participating in the DR) in multiple copies (to be resilient against harddisk failures). The aggregator then transfers this data to the TSO (to enable it to make payment decisions in DR settlement) and to a trusted third party/CA. Further, in case of a

³on successful verification of signature using corresponding public key.

TABLE I
ALL POSSIBLE CASES OF ARBITRATION. YES INDICATES THE CONCERNED PARTY IS ABLE TO PROVIDE SIGNED DATA, NO INDICATES OTHERWISE.

Third Party/CA	TSO	Aggregator	Arbitration
Yes	Yes	Yes	Yes
Yes	Yes	No	Yes
Yes	No	Yes	Yes
Yes	No	No	Yes
No	Yes	Yes	Yes
No	Yes	No	Yes
No	No	Yes	Yes
No	No	No	No

dispute, the payments from the escrow accounts are made based on the audit of the signed data by the trusted third-party/CA. Given the architecture of the SE and hardware integrity of smart meter, a malicious party cannot generate fake data for the same timestamp range with proper signature thereby making the dispute-solving procedure reliable⁴. This also ensures the arbitration in case of a dispute when at least one of the parties (CA/TSO/Aggregator) can provide the complete data.

Table I lists all the possible arbitration cases. It fails only if all the parties fail concurrently to provide the signed data. Next, unlike the referenced works in the Sec. II, we have an understanding that storing this signed power consumption data on the blockchain does not serve any additional benefit because this data is already tamper-proofed (by signing it using SE) and concerned parties will anyways store it for their protection of self-interest during a DR session. Moreover, as blockchain replicates everything on each participating node, this would generate unnecessary copies of this huge data-set(s) thereby causing scaling issues.

E. Synchronization in the Framework

This subsection deals with clock synchronization and tx pool synchronization in our framework. Starting with clock synchronization, during the DR session, tier 1 participants accept bids from aggregators (tier 2). These bids are only accepted between a given DR time window. If the tier 1 participants are not clock synchronized, then the gate closing time wouldn't be uniform across them. As aggregators are free to choose any tier 1 participant(s), clock synchronization is vital to avoid any unforeseen leverage (due to uneven gate closure time). As tier 1 participants are distributed and may in some cases belong to different geographical locations, an accurate time synchronization protocol is needed. PTPv2 [29] is one such protocol that can synchronize all the nodes within the sub- μ s range. PTPv2 is a layer 2 protocol meaning nodes outside a given subnet cannot be clock synchronized. To resolve this, a GPS clock is added as a grandmaster clock on each tier 1 participant, which PTPv2 uses to synchronize timing on them. The effective clock accuracy with this combined approach is in sub- μ s range as well⁵. This clock synchronization accuracy is acceptable considering that even real-time systems have sub- m s requirement.

Next, for the tx pool, synchronization to have the same sorted order of all bids is imperative to ensure the same input

⁴similar concept used in chip and pin credit/debit cards to secure offline tx.

⁵GPS clock accuracy is in sub- μ s range.

and thereby the same output for the DR allotment model when running in distributed setting across all tier 1 nodes. The sorting mechanism for bids in the common tx pool involves bid's hash. Here, we propose to use *SHA-512* hash function. Bid with the lowest magnitude of hash is ranked first and the rest are sorted similarly. If an aggregator sends the same bid to more than one tier 1 participant, then in the common tx pool, only one bid is kept (duplicates are removed). This ensures, for each DR session, there are only as many bids as the number of aggregators participating in that specific session. Further, In the case of hash collision i.e., the same hash for bids from two different aggregators, the bid with the lower hash value of *bid + mac address of the corresponding aggregator* is given precedence. As hashing is an unpredictable random function, fairness is thus ensured.

Further, to be deterministic with maximum delays while solving the DR allotment model (exponential time), there is a cap on the number of bids/aggregators on which the model is applied. Here again, the first n bids (thus, n aggregators considering 1 bid/aggregator) from the sorted synchronized tx pool are considered. The proper limit on n is elaborated in Sec. VI. Independently, Tier 1 participants also time-stamp the bids they receive. With this, there is a possibility that it is changed to favor a specific aggregator. To minimize the impact of this misbehavior, we propose to use time-stamp only for checking if the bid was received before the gate closure and not in the actual DR allotment model. Using this approach, it is impossible for any malicious tier 1 node to guarantee a favor by changing the timestamp for the colluding aggregator as the ordering is completely random (based on hash).

IV. DR ALLOTMENT MODELLING

Our framework is completely agnostic for the DR allotment model. Participants from tier 1 can plug in any DR allotment model in the framework which then has to be certified by CAs (see Sec. III-B) before being utilized in a DR session. There are DR allotment models proposed in previous works like [30], [31], etc., but they are adapted to micro-grid needs. For the completeness of our work and simplicity in understanding, we present a novel DR allotment model for TSO managed DR in the specific context of macro-grids. To determine an optimal DR allotment for the participating aggregators, the framework relies on a mixed-integer linear programming (MILP) formulation where integer variables are binary. More precisely, the MILP optimization tries to select a higher subset of aggregators while increasing the accepted fraction of their proposed bids (with respect to the DR signal), with corresponding higher bid amounts and reliabilities.

Each aggregator is assigned an average reliability value, computed by CAs/regulators and TSOs. The reliability of an aggregator defines its probability to satisfy a submitted bid. For example, a reliability of 50% for an aggregator means that it can, on average, satisfy half of its submitted bid. Initially, all aggregators have the same average reliability value, 100% which is then subsequently updated after each DR session by the TSO and is then stored on the blockchain (see Sec. V).

The total reliability of a subset of aggregators is the product of their reliabilities. Consequently, the logarithm of the total reliability is the sum of individual logarithmic reliabilities. The combined reliability for the selected aggregators should be greater than the TSO's required reliability. Thus, the MILP can be formulated as follows:

$$\begin{aligned} \text{Maximize: } & \frac{\alpha}{D} \sum_{i=1}^n f_i b_i r_i + \frac{\beta}{n} \times \sum_{i=1}^n [f_i] \\ \text{Subject to: } & \sum_{i=1}^n f_i \times b_i \leq D; \quad \sum_{i=1}^n [f_i] \times \log r_i \geq \log R; \\ & f_i \in \{0\} \cup \left[\max \{f_{i_{\min}}^{req}, f_{i_{\text{fair}}}\}, 1 \right], i = 1, \dots, n; \\ & [f_i] \in \{0, 1\}, i = 1, \dots, n \end{aligned} \quad (1)$$

Where:

- α, β are the scaling factors to balance DR value fulfillment and distribution;
- D is the total demand released by TSO;
- n is the total number of aggregators participating in a given DR session;
- R is the total reliability required by the TSO.

For aggregator i :

- b_i, r_i is the proposed bid of power consumption reduction and the avg. reliability respectively;
- f_i is a **variable** which denotes the fraction allotted by TSO to the aggregator i after optimization. Hence $f_i \times b_i$ is final power reduction to be done by aggregator i ;
- $[f_i]$ is the ceiling of f_i implemented by introducing a **binary variable** i.e., $w_i = [f_i] = 1$ if $f_i > 0$, else 0;
- $f_{i_{\min}}^{req}$ is the minimum fraction requested for allotment by aggregator i ;
- $f_{i_{\text{fair}}}$ is the fair fraction of allotment which ensures fair distribution of the DR signal and is given by:

$$f_{i_{\text{fair}}} = \frac{b_i \times r_i}{\sum_{j=1}^n b_j r_j}$$

As Equation 1 contains binary variables ($[f_i] \in \{0, 1\}$), it is categorized as NP-complete problem [32] due to the combinatorial explosion of its integer variables. Thus, for given n aggregators, there are at maximum 2^n possible distinct binary solutions where each solution is a subset of selected aggregators. There is no known polynomial-time algorithm to find an optimal solution. The exact algorithms (branch-and-bound, cutting-planes, etc.) that find an optimal solution may need an exponential number of iterations. As TSOs require an optimal solution, we apply the well known branch-and-bound method [33], as it reduces the number of iterations by exploring a subset of feasible integer solution instead of all possible solutions. Next, to be deterministic about maximum execution time, we propose a limit on the number of aggregators (n) for which the optimization algorithm will be applied (see Sec. VI).

V. DR FRAMEWORK IN ACTION

The DR session starts when the TSO multi-casts the DR requirement D using OpenADR [34]. All the subscribed

participants in the tier 1/tier 2 get notified about it. The gate for bidding is then opened by all participants in tier 1 (see Fig. 3). Interested aggregators in tier 2, send their secure signed (by SE) bid b_i to one or more tier 1 participants. This way, the acceptance of bids is completely decentralized and distributed thereby eliminating Single Point of Failure. When the gate closes, bids collected by all tier 1 participants are synchronized and sorted in the synchronized tx pool so that every tier 1 participant has the same set of sorted bids.

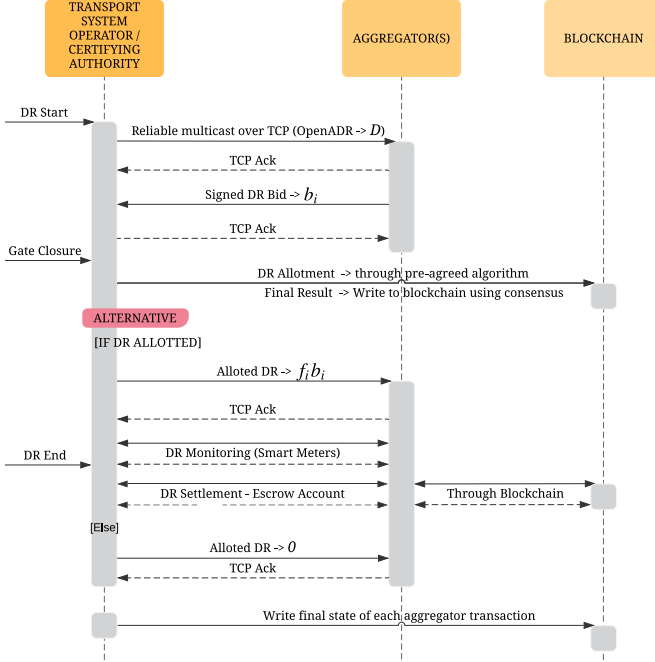


Fig. 3. A typical DR session overview.

Next, the certified DR allotment model (retrieved from the blockchain) is then applied on the common sorted bid list by each participant in tier 1. When the DR allotment is done and subsequent consensus among tier 1 participant is reached, the resulting DR allotment is written on the blockchain by adding a new block to the blockchain which contains all the bids but segregated into two groups viz., selected and rejected. This helps to maintain an immutable chronological history of the DR marketplace thus giving additional details such as the total energy bids accepted, current bidding trends, etc. This also makes it impossible for a malicious TSO to favor a particular aggregator as its proposed DR quota-distribution would be ruled-out during consensus.

Next, the communication between aggregator(s) and TSO is separate. Hence, delay by any aggregator to take required action does not affect the DR session for other remaining aggregators. Further, for each aggregator whose bid is accepted, TSO notifies the allotted percentage of its bid also pays in advance to the corresponding escrow account on the blockchain (see Sec. III-C). This ensures that DR participants are paid if they follow the allotted power consumption reduction baseline. If the TSO diverges from this assumed hypothesis, a proper redressal mechanism is to be used whose discussion is out of the scope of this paper.

Further, each selected aggregator is required to pay an advance fine in the corresponding escrow account (before the pre-agreed timeout) ensuring it is fined if it is not respecting allotted baseline. If the aggregator fails to prepay the fine before the time-out, it is not considered for the subsequent steps. DR monitoring starts only after prepayment from both sides. The DR monitoring is secured using smart meters and SE (see Sec. III-D). After DR monitoring ends, for DR settlement, TSO makes the payment/deduction using the same escrow account through blockchain and if aggregator disagrees, an arbitration is opened with a pre-agreed trusted third party/CA. After DR settlement ends, the final state of each DR bid along with updated parameters like reliability, response time, etc., for the corresponding aggregator is written on the blockchain.

VI. SIMULATION RESULTS

The DR allotment model proposed in Sec. IV was codified in python and used as ASC. PuLP library was used to solve the optimization problem. One example simulation is presented below. The required TSO parameters are given in Table II while parameters for individual aggregators are given in Table III.

TABLE II
PARAMETERS FOR TSO.

Parameter	α, β	D (MW)	$R(\%)$	n
Value	1	100	0.0001	5

TABLE III
INDIVIDUAL SIMULATION PARAMETERS FOR 5 AGGREGATORS.

Aggregator(i)	1	2	3	4	5
b_i (MW)	100	100	100	100	100
$f_{i_{min}}^{req}$	10%	10%	20%	25%	30%
r_i	10%	10%	20%	25%	30%

The final allotted DR quota (using the proposed DR allotment model) for each aggregator is shown in Fig. 4. The blockchain consensus, in this case, is very important as that makes the whole DR allotment decentralized. Once the optimization is solved by the TSO, it shares the results in a standard blockchain tx containing all the individual DR allotments. The dissemination happens using the blockchain P2P network maintained between all the tier 1 participants. One miner/verifier is chosen using the underlying consensus mechanism which will add this tx to a new block after verifying the results in tx by comparing it with locally obtained results. Then the block dissemination happens and every tier 1 entity similarly verifies the block before adding it to its local copy of the blockchain. Subsequently, tier 2 participants can access these results once they are on the blockchain making the whole process transparent.

Attack Model: Consider, TSO colludes with aggregator 1 to favor it with more allocation, as presented in Fig. 5. The presented DR quota satisfies all the constraints of Table II and III but still favors aggregator 1 with more DR quota. The TSO subsequently creates a standard blockchain tx containing this allotment scheme. However, during the consensus process, the tx will be invalidated because it won't match with locally

obtained results of other tier 1 entities making the allocation null and void. Instead, the proper allocation (Fig. 4) will be validated as long as the majority of the network ($> 50\%$) validates it.

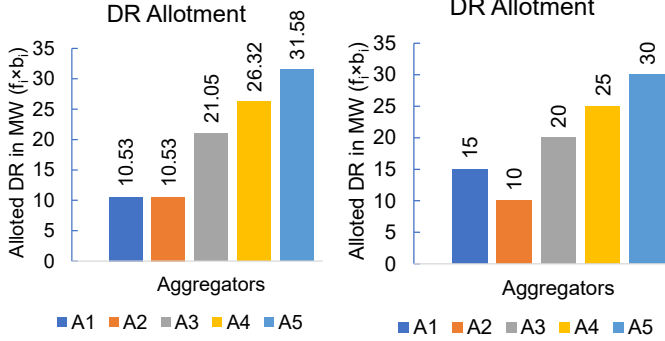


Fig. 4. DR allotment using the proposed model.

Fig. 5. Biased DR allotment by TSO favoring aggregator 1.

Further, the performance of the DR allotment model proposed in Sec. IV was evaluated against a wide range of simulated inputs and the results shows the effect of increasing the number of aggregators against various chosen system reliabilities in four areas, i.e., 1) Percentage of accepted bids, 2) Percentage of accepted aggregators 3) Average percentage allotment for selected subset of aggregators and 4) Time required to complete the allotment. For simulations, parameters from Equation 1 are set as given in Table IV for TSO and in Table V for aggregators. The averaged results for 1000 iterations are presented next.

TABLE IV
PARAMETERS FOR TSO.

Parameter	α, β	D (MW)	$R(\%)$	n
Value	1	100	5, 25, 50, 75, 95	[10, 50], step=10

TABLE V
PARAMETERS FOR AGGREGATORS.

Parameter	b_i (MW)	$f_{i,min}^{req}$ (%)	r_i (%)
Value	$random(0, 50)$	$random(0, 100)$	$random(0, 100)$

Fig. 6 shows the percentage of fulfilled DR. It is evident that irrespective of the value of R , the fulfillment percentage increases due to an increase in n which subsequently increases participants with required R . However, higher R requirement ($= 95\%$) can reduce it to meagre 29% (on average, for lower n). Fig. 7 shows that the % of selected aggregators is greatly reduced when total participating aggregators (n) is increased as the demand D stays constant. The negative slope (< -1) of curves indicates that indeed the proposed MILP finds a delicate balance between distributing the demand and respecting the minimum allotment.

Fig. 8 shows that the avg. allotted fraction f_i gradually reduces with increase in n or decrease in R . This is because both trends (increasing n , decreasing R) increase the number of actual aggregators being considered and hence the demand D gets divided. Next, Fig. 9 depicts the execution time taken by the DR allotment model for various values of (n, R) which shows that it is directly dependent on the actual number of

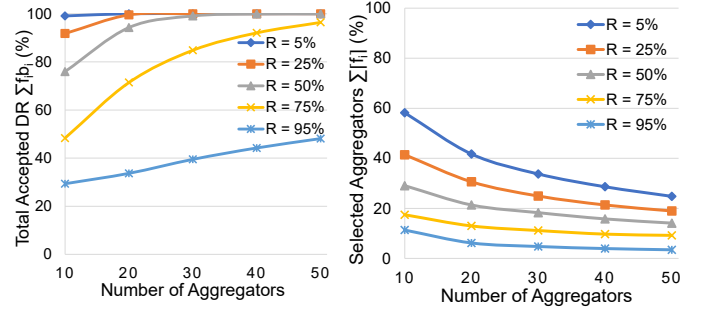


Fig. 6. Total accepted DR (percentage of fulfillment).

Fig. 7. The average percentage of selected aggregators.

aggregators under consideration and can be reduced by as much as 82% by increasing R from 5% to 95% (on average for higher n). Further, for the lower value of R , the execution time rises more drastically ($\approx 4.5x$) for an increase in n .

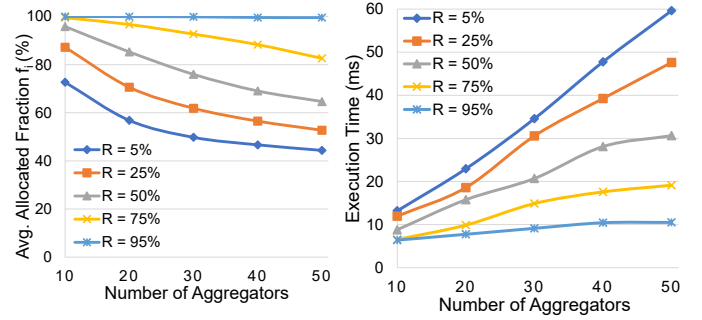


Fig. 8. Average allotted fraction (f_i) for selected aggregators.

Fig. 9. Execution time taken by the DR allotment model.

To summarize, with the proposed DR allotment model, a very low or very high value of R drastically affects some or the other metrics of DR allotment. Owing to this and to be deterministic about maximum execution time, we propose a suitable limit of $n = 20000$ as the MILP for it is solved within 55 seconds on i7-8550U CPU (1.80GHz). Considering real-world application, $n = 20000$ aggregators limit is very high, thus, making it apt for future scaling needs. Subsequently, the inter-block time for block formation in the underlying blockchain should also be higher than 55 seconds (e.g. 2 minutes).

VII. CONCLUSION AND FUTURE WORK

In this paper, we proposed a blockchain-based framework for Energy DR bringing in more transparency, trustlessness, non-repudiation and introducing complete decentralization of all the activities of a DR session while meticulously dodging blockchain issues like scaling, SC overhead, etc. Further, we proposed an efficient DR allotment model by formulating an optimization MILP. The simulations showed that even for n as high as 20000, the solving time is less than 1 minute. Further, it showed that a very low or very high value of required system reliability has an inordinate effect on different DR allotment metrics. Our future work will be to consider security analysis of blockchain (like Byzantine faults, forks, etc.) related to our framework and proposing an upgraded MILP to finely control the effects of individual bids and reliabilities.

REFERENCES

- [1] S. Haber and W. S. Stornetta, "How to time-stamp a digital document," in *Conference on the Theory and Application of Cryptography*. Springer, 1990, pp. 437–455.
- [2] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Bitcoin*, 2008.
- [3] R. Alexander, *IOTA-Introduction to the Tangle Technology: Everything you need to know about the revolutionary blockchain alternative*. Independently published, 2018.
- [4] T. McConaghy, R. Marques, A. Müller, D. De Jonghe, T. McConaghy, G. McMullen, R. Henderson, S. Bellemare, and A. Granzotto, "Bigchaindb: a scalable blockchain database," *white paper, BigChainDB*, 2016.
- [5] H. M. Kim and M. Laskowski, "Toward an ontology-driven blockchain design for supply-chain provenance," *Intelligent Systems in Accounting, Finance and Management*, vol. 25, no. 1, pp. 18–27, 2018.
- [6] C. Zhang, J. Wu, C. Long, and M. Cheng, "Review of existing peer-to-peer energy trading projects," *Energy Procedia*, vol. 105, pp. 2563–2568, 2017.
- [7] I. Linkeviciute, "Essays on the demand-side management in electricity markets," *Copenhagen Business School, Denmark*, 2019.
- [8] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini, "Blockchain based decentralized management of demand response programs in smart energy grids," *Sensors*, vol. 18, no. 1, p. 162, 2018.
- [9] S. Noor, W. Yang, M. Guo, K. H. van Dam, and X. Wang, "Energy demand side management within micro-grid networks enhanced by blockchain," *Applied energy*, vol. 228, pp. 1385–1398, 2018.
- [10] K. Gai, Y. Wu, L. Zhu, M. Qiu, and M. Shen, "Privacy-preserving energy trading using consortium blockchain in smart grid," *IEEE Transactions on Industrial Informatics*, 2019.
- [11] X. Wu, B. Duan, Y. Yan, and Y. Zhong, "M2m blockchain: The case of demand side management of smart grid," in *2017 IEEE 23rd International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 2017, pp. 810–813.
- [12] G. Greenspan, "Multichain private blockchain, white paper," *URL: <http://www.multichain.com/download/MultiChain-White-Paper.pdf>*, 2015.
- [13] F. Imbault, M. Swiatek, R. De Beaufort, and R. Plana, "The green blockchain: Managing decentralized energy production and consumption," in *2017 IEEEIC/I & CPS Europe*. IEEE, 2017, pp. 1–5.
- [14] V. Deshpande, H. Badis, and L. George, "Btcmmap: Mapping bitcoin peer-to-peer network topology," in *2018 IFIP/IEEE International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN)*, Sep. 2018, pp. 1–6.
- [15] V. Deshpande, L. George, and H. Badis, "Safe: A blockchain and secure element based framework for safeguarding smart vehicles," in *2019 12th IFIP Wireless and Mobile Networking Conference (WMNC)*, Sep. 2019, pp. 181–188.
- [16] C. Cachin, "Architecture of the hyperledger blockchain fabric," in *Workshop on distributed cryptocurrencies and consensus ledgers*, vol. 310, 2016, p. 4.
- [17] C. Dannen, *Introducing Ethereum and Solidity*. Springer, 2017.
- [18] Monax, "Hyperledger burrow," The Linux Foundation. [Online]. Available: <https://github.com/hyperledger/burrow>
- [19] F. Hasse, A. von Perfall, T. Hillebrand, E. Smole, L. Lay, and M. Charlet, "Blockchain—an opportunity for energy producers and consumers," *PwC Global Power & Utilities*, pp. 1–45, 2016.
- [20] S. Floyd, V. Jacobson, C.-G. Liu, S. McCanne, and L. Zhang, "A reliable multicast framework for light-weight sessions and application level framing," *IEEE/ACM transactions on networking*, no. 6, pp. 784–803, 1997.
- [21] C. Diot, W. Dabbous, and J. Crowcroft, "Multipoint communication: a survey of protocols, functions, and mechanisms," *IEEE journal on selected areas in communications*, vol. 15, no. 3, pp. 277–290, 1997.
- [22] C. Banton, "Escrow: What is escrow?" Investopedia, 2019. [Online]. Available: <https://www.investopedia.com/terms/e/escrow.asp>
- [23] LocalEthereum. [Online]. Available: <https://localetereum.com/>
- [24] M. I. Mehar, C. L. Shier, A. Giambattista, E. Gong, G. Fletcher, R. Sanayhie, H. M. Kim, and M. Laskowski, "Understanding a revolutionary and flawed grand experiment in blockchain: the dao attack," *Journal of Cases on Information Technology (JCIT)*, vol. 21, no. 1, pp. 19–32, 2019.
- [25] GlobalPlatform, Inc., "Introduction to secure element," 2018.
- [26] V. Deshpande, L. George, and H. Badis, "Pulsec: Secure element based framework for sensors anomaly detection in industry 4.0," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 1204 – 1209, 2019, 9th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2019.
- [27] V. Deshpande, T. Das, H. Badis, and L. George, "Sebs: A secure element and blockchain stratagem for securing iot," in *2019 Global Information Infrastructure and Networking Symposium (GIIS)*, Dec 2019, pp. 1–7.
- [28] C. Adams and S. Lloyd, *Understanding public-key infrastructure: concepts, standards, and deployment considerations*. Sams Publishing, 1999.
- [29] IEEE Instrumentation and Measurement Society, "IEEE Std1588TM2008," 2008.
- [30] A. J. Conejo, J. M. Morales, and L. Baringo, "Real-time demand response model," *IEEE Transactions on Smart Grid*, vol. 1, no. 3, pp. 236–242, 2010.
- [31] L. Chen, N. Li, S. H. Low, and J. C. Doyle, "Two market models for demand response in power networks," in *2010 First IEEE International Conference on Smart Grid Communications*. IEEE, 2010, pp. 397–402.
- [32] R. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, R. Miller and J. Thatcher, Eds. Plenum Press, 1972, pp. 85–103.
- [33] A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems," *Econometrica*, vol. 28, no. 3, pp. 497–520, 1960. [Online]. Available: <http://www.jstor.org/stable/1910129>
- [34] OpenADR Alliance, "OpenADR 2.0 profile specification," *document 20120912-1*, 2013.