Multi-Agent Architecture for Peer-to-Peer Electricity Trading based on Blockchain Technology

Yeray Mezquita,
Amin Shokri Gazafroudi,
J. M. Corchado
BISITE Research Group
University of Salamanca
Salamanca, Spain
{yeraymm, shokri}@usal.es,
corchado@usal.es

Miadreza Shafie-khah,
Hannu Laaksonen
School of Technology and Innovations
University of Vaasa
Vaasa, Finland
mshafiek@univaasa.fi,
hannu.laaksonen@uwasa.fi

Aida Kamišalić
Faculty of Electrical Engineering
and Computer Science
University of Maribor
Maribor, Slovenia
aida.kamisalic@um.si

Abstract—The security of smart grids is put at risk due to their automation and remote access features. Blockchain technology can be used as a distributed ledger where data is stored and all the data transactions between the different entities of a smart grid are signed to protect them from such attacks. This paper proposes a multi-agent system (MAS) that combines smart contracts and blockchain to enable Peer-to-Peer electricity trading in a MicroGrid (MG) scenario, without the need for human intervention. The use of blockchain technology helps reduce transaction costs and allows to make micro transactions in the proposed market. Blockchain also improves the security of the platform because all the involved actors can be certain about the authorship of the information produced in the system. Finally, the use of a MAS and the possibility of negotiating between the agents helps obtain an optimal state in the system in which the costs of energy are minimal and the local production of energy is profitable.

Index Terms—Blockchain, local electricity market, multi-agent system, peer-to-peer, smart grid

I. Introduction

In the future, Information and Communication Technologies will become substantial enablers of new, cost-effective Smart Grid control functionalities. However, this development also entails some risks. The control and management of the grid may become vulnerable to security attacks and the supply of electricity may become unreliable [1].

In order to protect the smart grid, Blockchain Technology (BT) can be used to protect the smart grid; BT acts as a reliable distributed ledger for the storage of data. Additionally, this technology provides a mechanism to sign and encrypt the interactions between the different actors of the platform [2].

According to the security mechanism of BT, data is signed and encrypted with pairs of private and public keys. Thus, if the BT used is secure enough [3], as long as the keys are not compromised, the system will be protected from hackers, ensuring the integrity and authorship of the data through the

users' signatures [4]. Also, the data stored in the blockchain is tamper-proof, it remains in that state permanently and cannot be modified [5].

Another feature of BT-based systems is their ability to make automatic micropayments between machines, like smart meters, without the need for human intervention [6]. Micropayments are automatic when certain conditions, previously agreed upon by the stakeholders, are met [7]. Smart contracts are the programs stored in the blockchain that are responsible for ensuring these conditions are complied with [8].

There are several works that discuss the use of BT in power and energy systems. Pieroni et al. [9] proposed an innovative synergy between BT and microgrids (MGs), applying it within the scope of smart cities. BT can play an important role in the MG, in terms of communications, transactions and the security of the stakeholders. The combination of both technologies allows for the implementation of a local energy market between the actors that produce/consume energy.

In order to optimize the balance between energy demand and energy generation, Noor et al. [10] proposed a game theory-based model for demand side management. It is a non-cooperative game between actors, that take part in the energy MG platform, in order to optimize their own payoffs. In this case, Blockchain was used as a distributed control centre for information exchange, creating a transparent energy market. It has been probed through a simulation that this kind of approach is applicable to real world applications [10].

A constrained non-linear programming optimization algorithm with rolling horizon was proposed by Long et al. [11] in order to minimize the energy costs of the actors that participate in the MG. Thanks to the use of BT, it is possible for peers to share energy directly with each other. Authors proposed a method of peer-to-peer energy trading that reduces the cost of the energy by 30%, in comparison to the conventional Peer-to-Grid method [11].

The general trend in research is to develop models where MGs benefit from BT features, such as decentralization and security [12], [13], [14], [15].

After studying the performance and the viability of state-

of-the-art BT-based systems, we propose in this article an architecture of a distributed Multi-Agent System (MAS) which makes use of the Ethereum blockchain, to enable decentralized control over the platform. The MAS paradigm has been chosen because previous researches have demonstrated that platform performance can improve significantly thanks to a MAS architecture [16]–[23].

In the proposed system, the MAS is in charge of correct MG management and energy trade between peers within the local market. Communication between the agents is also coordinated through the blockchain, as they work on the smart optimization of the costs and payoffs of the agents that participate in the system. This approach makes it possible to deploy a totally decentralized platform, without a single-point of failure.

Section II describes the proposed architecture which combines BT and the MAS paradigm for improved security, creation of a local market, and optimization of the payoffs for the MG actors/stakeholders. The fundamental stages in the deployment and implementation of the system are outlined in Section III. Section IV draws conclusions and final remarks on the conducted research.

II. PROPOSED ARCHITECTURE

As shown in Fig. 1, the proposed system consists of three separate parts: the device layer, composed mainly of smart meters, Photo-Voltaic (PV) panels and batteries; the MAS, which is implemented in the Python Agent DEvelopment (PADE) framework with Python 3.6; and the selected BT, in this case the Ethereum blockchain and network have been used.

Smart meters make it possible to read the amount of energy consumed by different domestic appliances. In order to enable principal devices to interact with the blockchain, each of them is connected to an Arduino ESP32 module. This module is connected to the internet too. Also, a private library has been developed which enables the interaction of the devices with the Ethereum blockchain. Thus, the states and measures taken by the different devices are written in the blockchain.

BT is central to the communications of the platform. All information related to the states of the MAS agents is stored in the blockchain. Moreover, the different transactions carried out between agents are logged there.

The MAS is in charge of the management and control of the MG. The negotiations carried out between the agents help the consumer agents reduce the cost of the energy they need, while improving the payoffs of the seller agents, obtaining a balance in the price of the energy from which all the actors in the system benefit.

A. Blockchain technology

In the proposed platform, the energy is exchanged in batches. Each hour the consumer agents start a negotiation process to buy an estimated quantity of energy they will need for the next hour. In this way, the time confirm the new data stored in the blockchain is not a problem [24]. The consumer agents will make use of 50 minutes of that hour, to negotiate
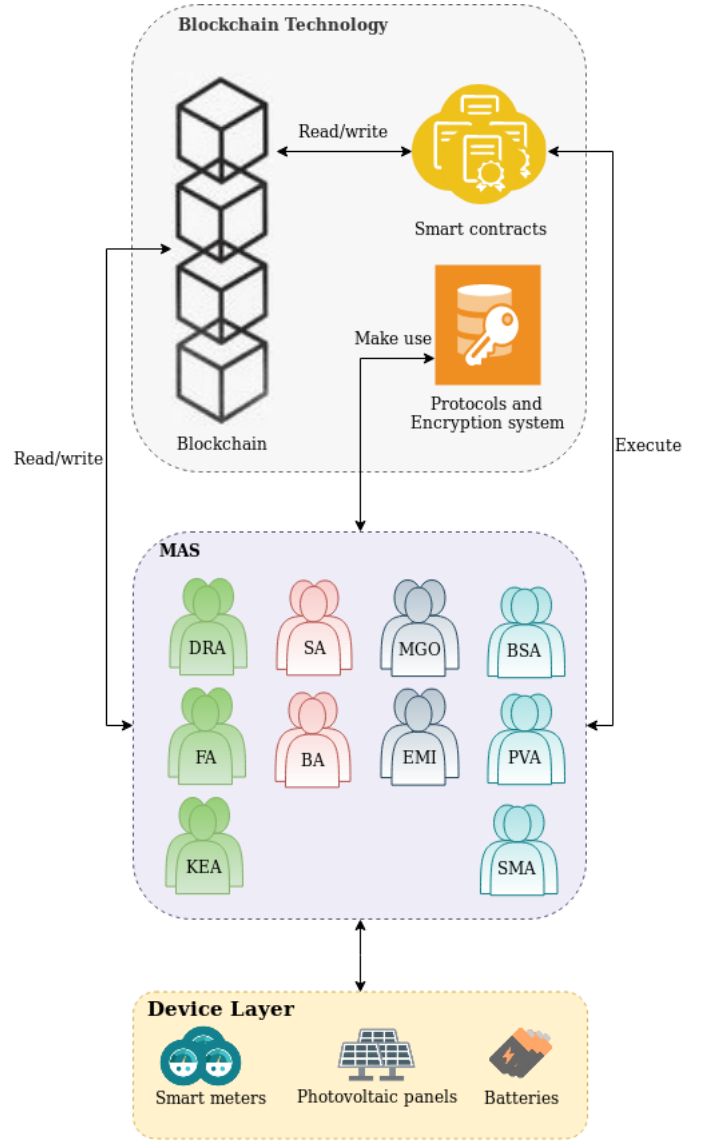

Fig. 1. Proposed system architecture.

with the producers. The last ten minutes is enough time to confirm the new data added by the smart contracts to the blockchain.

Public blockchains offer solid functionalities; they give their users the certainty that the data they store in the blockchain will be kept tamper-proof. However, in order to maintain these networks alive, it is necessary to come up with some kind of incentive for individual nodes. Ethereum Gas is the incentive that keeps the Ethereum Blockchain running. Gas is the Interface of the cyptocurrency native of Ethereum (Ether), which means that the value of Gas depends on Ether. This currency is paid by the user to the network when the user executes a function of a smart contract that changes its state. According to [25], to execute a smart contract in Ethereum, a fixed amount of 21000 Gas is needed plus the amount of Gas need to execute smart contract operations. If the price of Gas

as of 23-08-2017 was 28 Gwei, then, the execution price of a transaction is equivalent to 0.1€. In this platform, the average amount of transactions established by the proposed protocol is around 4, with a price of 0.4€:

- 1 transaction to publish the offer of watts in the blockchain.
- 1 transaction to update the offer of watts when sold to a consumer.
- 1 to approve the amount of currencies to be spent by the smart contract from the consumer to the producer after an agreement is reached.
- 1 transaction to give the currencies to the producer previously approved by the consumer from the smart contract.

So, in order to make use of the Ethereum blockchain, each negotiation has to take into account the amount of transactions that must be performed, adding their cost to the minimum price a batch of watts must be sold. Another requisite of the system is that the devices that interact with the blockchain must have internet access to use its services.

To test this system, the Rinkeby network is used. Rinkeby is a test network based on Ethereum that makes use of Proof of Authority (PoA) as the consensus algorithm. This consensus algorithm is based on the fact that only approved accounts can become validators. Their role is to validate the transactions carried out in the network and store them inside the blockchain. It is a more centralized solution than other types of consensus algorithms, but it is immune to spam.

In this work, the main goal is to test if it is viable this kind of platform in the real world. Although Rinkeby doesn't offer the same security characteristics of Ethereum, it is valid to use it in this simulation, because it has been used as if it have the same validation time and costs of Ethereum.

Only verifiable actors can participate in the proposed platform. If new actors, like smart meters or PV panels, want to enter the platform and be part of it, they are added by agents responsible for their installation in the platform. This way, it is more difficult for a malicious actor to enter the platform and disrupt its well functioning.

The energy traded in the platform is virtualized in the blockchain following the fungible Ethereum ERC20 token standard. This kind of standard is chosen, because all the energy blocks have the same characteristics.

To be valid, the energy token must meet the following requirements:

- Guaranteed: Each token must be guaranteed by the correct quantity of energy that backs it up.
- Auditable: There have to be audits of the energy that backs up the tokens stored in the blockchain. The smart meters automatically write in the blockchain the energy that passes through them, triggering a smart contract when a block of energy reaches its destination, and executing the payment of the consumer to the producer. The elements of the platform which insert data from the real world into the blockchain are called oracles. In this platform, the ESP32 modules are the ones with that role, storing the data the sensors associated to them are reading from the real world. They only insert the data that are required by a smart contract [26].

For that reason, the mentioned elements, should be deployed on the platform sealed. Periodic audits from those elements are needed to guarantee the existence of the assets and the accuracy of the oracles, while detecting malicious actors that want to change the behaviour of the audited elements.

After coding the energy token based on the ERC20 token standard and another ERC20 token to simulate the fiduciary money; a smart contract acts as the intermediary between the agents, enabling them to trust in the agreement reached by the agents through the negotiation process. The main functions of the coded smart contracts include:

- PublishInfo: A function that lets the actors of the system store and update their information in the blockchain.
- GetInfo: A function that lets other agents read the information published by a given agent.
- MakeOffer: A function that puts a producer's energy blocks on sale.
- AllocateEnergy: A function that allocates an amount of the producer's energy to a given consumer for a given price.
- AllowTransaction: A function that puts the given amount of virtualized fiduciary money in the smart contract.
- MakeTransaction: Once the system has detected that the purchased amount of energy has been transacted, the smart contract gives the virtualized fiduciary money from the consumer to the producer.
- MintWatts: Approved actors can mint watts in order to virtualize the quantity of energy producers have stored. Producers ask for the mint of watts every 15 minutes and if they have stored a enough watts to make profitable their virtualization on the blockchain.

Each ESP32 module have a wallet with its own ERC20 tokens representing the fiduciary money and Ether to pay for their transactions. That money is minted to those accounts by approved actors when they receive real fiduciary money from other actors. Also, the actors in charge of the mint are in charge of burning fiduciary tokens when they are asked to return the fiduciary money to its owner.

Regarding the security of the platform, each ESP32 module stores its own private key. So, as long as it continues sealed, its use will continue to be safe.

B. Multi-agent system

The proposed MAS is developed in the PADE framework. Four different subsystems are defined in the proposed MAS (see Fig. 1). Each MAS subsystem is in charge of a single functionality of the platform.

The Device Driver System (DDS) groups the agents which control the smart devices. Those agents are deployed inside the ESP32 modules depending on the type of devices they are

associated with. In this way, they interact with different smart contracts to store their corresponding type of data. The Smart Meter state Agents (SMA) are in charge of storing the state and the energy that flows through the smart meters. The PV state Agents (PVA) monitor the state of the PV panels (their production of energy, active and reactive powers, voltage and current), and storing them in the blockchain. The Battery State Agents (BSA) are associated with the batteries of the system in order to store their current state of charge, their charged and discharged powers, in the blockchain.

The Transaction Manager System (TMS) groups the agents that are in charge of the Peer-to-Peer energy trading. The Buyer Agent (BA) searches the energy offers and begins the negotiation process with its producers. The Seller Agent (SA) calculates the tendency of energy price and its demand - with the help of the Data Analytic System (DAS) - and publishes offers in the blockchain when it detects a surplus of energy.

The Microgrid Operator System (MGOS) consists of a set of agents that monitor the state of the MG. The Microgrid Operator agent (MGO) monitors the MG network in order to manage the power balance, voltage level as well as power flows between the MG and the utility network, depending on the real-time electricity prices (local peer-to-peer market and wholesale electricity market prices) and state-of-charge (SOC) of microgrid battery storage(s). This agent also verifies the addition of new stakeholders in the platform. The External Market Interactuator agent (EMI) monitors the wholesale electricity market. In this way, the DAS predicts the electricity price and informs the MGO about it.

The DAS arranges the agents which will perform data analytics for the extraction of knowledge and allow for the predictions required by the other agents in the system. The Data Reader Agents (DRAs) read data from different sources like the blockchain or from external sources. The read and parsed data is used by the Knowledge Extractor Agents (KEAs) to constantly train new models for greater prediction accuracy. The Forecast Agents (FA) obtain the models created by the KEA and use them when requested to make a prediction, indicating what actions should be taken.

III. PLATFORM DEPLOYMENT

The participants that make up the platform can be buildings with the role of consumers, PV panels as producers, or batteries as prosumers. In Fig. 2, the deployment of the system is shown. Each participant of the system has its own set of agents which allows them to participate in the electricity market.

All the connections between the different parts of the system are done via Internet. The Ethereum Network is a network of autonomous nodes which is independent of the platform. It ensures that data are stored securely, in addition to acting as the central part of the communications in the system. The devices are controlled by the ESP32 modules. In each ESP32 module, an agent is deployed which depends on its associated device. These agents are in charge of reading the state of the
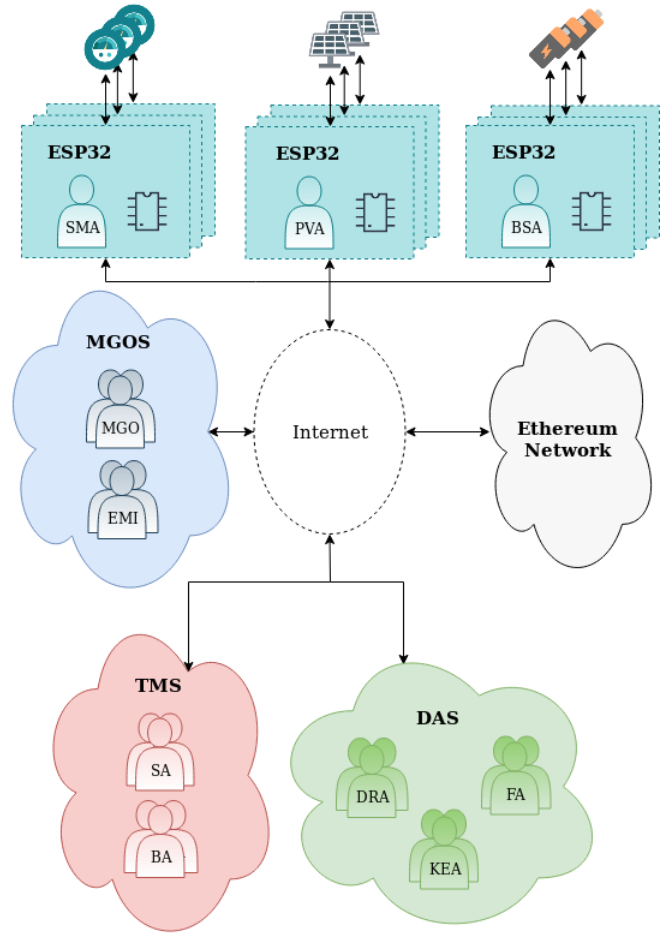

Fig. 2. Deployment of the platform.

corresponding devices and writing it in the blockchain through the internet.

The rest of the agents are deployed in the cloud. Thus, their server addresses are stored and updated in the blockchain to let the other agents know how to communicate with them. This provides versatility to the users who want to participate in the system, letting them deploy the agents in their own local servers or in environments like Google Cloud. In this way, if a participant is a consumer/producer, they do not need to deploy the BA/SA. In the case of the agents of the DAS, each participant can deploy its own BA/SA which does not depend on others.

The SA and BA read the state of the participants that are associated with them in the blockchain. If the SA detects that the PV panel's battery is going to be fully charged, it will tokenize an amount of energy with an initial cost of automatic adjudication. It determines the amount and the cost with the help of an FA, in order to obtain the best payoff from it on the basis of the offer and demand of the MG market. If the BA demands energy, it searches for tokenized energy offers in the blockchain and buys the cheaper ones until it meets its demand. If there is an unbalance of energy in the MG and it

is needed a supply of energy, the MGO injects it. Then the entities of the platform that need to consume that energy pay for it at the market price.

IV. Conclusions

This work has proposed a MAS-based architecture that integrates BT within it in order to allow peer-to-peer transactions of energy in a MG. The MAS is in charge of automating and optimizing the interactions between entities in this platform, following the Game-Theory model of negotiations proposed in [10].

Thus, agents manipulate the blockchain through smart contracts implemented in solidity, tokenizing and updating the information associated with the energy traded in the platform. The ERC20 standard is followed for fungible tokens and it is the standard used to virtualize the energy exchanged in the system as well as the virtualized real money.

Thanks to the use of BT, it is possible to create a transparent and efficient market of electricity, as studied in the literature review of this work, in which the peers trade electricity directly between them. Also, this approach makes use of the characteristics of the Ethereum blockchain and network of nodes, in order to have a decentralized tamper-proof registry which allows for the establishment of trustful agreements between agents, and the data recorded by them. Finally, it is the central mechanism of communications of the platform, used by the agents of the MAS to localize other agents and communicate with them.

The use of a blockchain as a central actor helps avoid single point of failure during the life cycle of the system. The platform also makes use of ESP32 Arduino modules to act as oracles of the state of the devices that work in the system: batteries, smart meters and PV panels. The deployment of devices and the storage of their state in the blockchain helps to audit the data stored in the blockchain. Additionally, the diffrent agents in the system use automatic decision making to optimize their payoffs in the electricity market. The implementation of smart contracts allows for the automation of energy and energy flexibility transactions between the actors in the platform.

We make use of a non-cooperative game theory model similarly to the approach of Noor et al. [10]. However, due to the nature of Ethereum, its verification speed and its fees [24], it is not possible to make use of the data produced in real time. Therefore, our approach use a batch of data obtained every hour. It has been selected that period of time to let the agents have more time to reach an agreement with lower prices. Also, thanks to that time window, stoping the negotiation process some minutes before its end, is enough time to let the Ethereum network confirm the changes made by the smart contracts after the negotiation process. Additionally, Long et al. approach [11] takes advantage of local batteries in which actors of the system can store their surplus energy.

The advantage of our proposal over the others, is in the consideration of technical details of the Ethereum blockchain and its limitations. In order to continue improving this work,

we will opt for usage of a different BT that does not require the payment of as many fees, in order to increase the payoffs of the agents. Another important aspect that we will focus on is reducing the time lapse between the storage of the different batches of data.

Also, thanks to the incorporation of the agents from the DAS, it is possible to create a local market, not only for energy, but for data and machine learning models. This market is not implemented in this work, but it will be in, as a future work.

References

[1] A. A. Memon and K. Kauhaniemi, "A critical review of ac microgrid protection issues and available solutions," *Electric Power Systems Research*, vol. 129, pp. 23–31, 2015.

[2] D. K. Tosh, S. Shetty, X. Liang, C. Kamhoua, and L. Njilla, "Consensus protocols for blockchain-based data provenance: Challenges and opportunities," in *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*. IEEE, 2017, pp. 469–474.

[3] V. Buterin, "Ethereum: Platform review," *Opportunities and Challenges for Private and Consortium Blockchains*, 2016.

[4] S. Huh, S. Cho, and S. Kim, "Managing iot devices using blockchain platform," in *2017 19th international conference on advanced communication technology (ICACT)*. IEEE, 2017, pp. 464–467.

[5] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla, "Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability," in *Proceedings of the 17th IEEE/ACM international symposium on cluster, cloud and grid computing*. IEEE Press, 2017, pp. 468–477.

[6] J. J. Sikorski, J. Haughton, and M. Kraft, "Blockchain technology in the chemical industry: Machine-to-machine electricity market," *Applied Energy*, vol. 195, pp. 234–246, 2017.

[7] K. N. Khaqqi, J. J. Sikorski, K. Hadinoto, and M. Kraft, "Incorporating seller/buyer reputation-based system in blockchain-enabled emission trading application," *Applied Energy*, vol. 209, pp. 8–19, 2018.

[8] I. Weber, X. Xu, R. Riveret, G. Governatori, A. Ponomarev, and J. Mendling, "Untrusted business process monitoring and execution using blockchain," in *International Conference on Business Process Management*. Springer, 2016, pp. 329–347.

[9] A. Pieroni, N. Scarpato, L. Di Nunzio, F. Fallucchi, and M. Raso, "Smarter city: smart energy grid based on blockchain technology," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 8, no. 1, pp. 298–306, 2018.

[10] 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.

[11] C. Long, J. Wu, Y. Zhou, and N. Jenkins, "Peer-to-peer energy sharing through a two-stage aggregated battery control in a community microgrid," *Applied energy*, vol. 226, pp. 261–276, 2018.

[12] 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.

[13] F. Imbault, M. Swiatek, R. De Beaufort, and R. Plana, "The green blockchain: Managing decentralized energy production and consumption," in *2017 IEEE International Conference on Environment and Electrical Engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*. IEEE, 2017, pp. 1–5.

[14] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, pp. 840–852, 2018.

[15] E. Mengelkamp, J. Gärttner, K. Rock, S. Kessler, L. Orsini, and C. Weinhardt, "Designing microgrid energy markets: A case study: The brooklyn microgrid," *Applied Energy*, vol. 210, pp. 870–880, 2018.

[16] M. Francisco, Y. Mezquita, S. Revollar, P. Vega, and J. De Paz, "Multi-agent distributed model predictive control with fuzzy negotiation," *Expert Systems with Applications*, 2019.

[17] A. S. Gazafroudi, F. Prieto-Castrillo, T. Pinto, and J. M. Corchado, "Organization-based multi-agent system of local electricity market: bottom-up approach," in *International Conference on Practical Applications of Agents and Multi-Agent Systems*. Springer, Cham, 2017, pp. 281–283.

[18] A. González-Briones, J. A. Castellanos-Garzón, Y. Mezquita Martín, J. Prieto, and J. M. Corchado, "A framework for knowledge discovery from wireless sensor networks in rural environments: A crop irrigation systems case study," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

[19] A. S. Gazafroudi, J. F. De Paz, F. Prieto-Castrillo, G. Villarrubia, S. Talari, M. Shafie-khah, and J. P. Catalão, "A review of multi-agent based energy management systems," in *International Symposium on Ambient Intelligence*. Springer, Cham, 2017, pp. 203–209.

[20] A. S. Gazafroudi, T. Pinto, F. Prieto-Castrillo, J. Prieto, J. M. Corchado, A. Jozi, Z. Vale, and G. K. Venayagamoorthy, "Organization-based multi-agent structure of the smart home electricity system," in *2017 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2017, pp. 1327–1334.

[21] A. S. Gazafroudi, F. Prieto-Castrillo, T. Pinto, A. Jozi, and Z. Vale, "Economic evaluation of predictive dispatch model in mas-based smart home," in *International Conference on Practical Applications of Agents and Multi-Agent Systems*. Springer, Cham, 2017, pp. 81–91.

[22] S. Najafi, S. Talari, A. S. Gazafroudi, M. Shafie-khah, J. M. Corchado, and J. P. Catalão, "Decentralized control of dr using a multi-agent method," in *Sustainable interdependent networks*. Springer, Cham, 2018, pp. 233–249.

[23] A. Shokri Gazafroudi, J. Prieto, and J. M. Corchado, "Virtual organization structure for agent-based local electricity trading," *Energies*, vol. 12, no. 8, p. 1521, 2019.

[24] C. Combi, "What are blockchain confirmations and why do they matter?" May 2017, [Online; updated 10-October-2018]. [Online]. Available: https://coincentral.com/blockchain-confirmations/

[25] D. Ryan, "Calculating costs in ethereum contracts," May 2017, [Online; updated 06-September-2019]. [Online]. Available: https://hackernoon.com/ether-purchase-power-df40a38c5a2f

[26] V. Gatteschi, F. Lamberti, C. Demartini, C. Pranteda, and V. Santamaría, "Blockchain and smart contracts for insurance: Is the technology mature enough?" *Future Internet*, vol. 10, no. 2, p. 20, 2018.