


Article

A Blockchain-Supported Framework for Charging Management of Electric Vehicles

Marina Dorokhova ^{1,*}, Jérémie Vianin ², Jean-Marie Alder ², Christophe Ballif ¹, Nicolas Wyrsh ¹
and David Wannier ²

- ¹ École Polytechnique Fédérale de Lausanne (EPFL), STI IMT PV-LAB, CH-2002 Neuchâtel, Switzerland; christophe.ballif@epfl.ch (C.B.); nicolas.wyrsh@epfl.ch (N.W.)
² Research Institute of Information Systems—Easilab, HES-SO Valais/Wallis, CH-3960 Sierre, Switzerland; jeremie.vianin@hevs.ch (J.V.); jean-marie.alder@hevs.ch (J.-M.A.); david.wannier@hevs.ch (D.W.)
* Correspondence: marina.dorokhova@epfl.ch

Abstract: Profound changes driven by decarbonization, decentralization, and digitalization are disrupting the energy industry, bringing new challenges to its key stakeholders. In the attempt to address the climate change issue, increasing penetration of renewables and mobility electrification augment the complexity of the electric grid, thus calling for new management approaches to govern energy exchanges while ensuring reliable and secure operations. The emerging blockchain technology is regarded as one of the most promising solutions to respond to the matter in a decentralized, efficient, fast, and secure way. In this work, we propose an Ethereum-based charging management framework for electric vehicles (EVs), tightly interlinked with physical and software infrastructure and implemented in a real-world demonstration site. With a specifically designed solidity-based smart contract governing the charging process, the proposed framework enables secure and reliable accounting of energy exchanges in a network of trustless peers, thus facilitating the EVs' deployment and encouraging the adoption of blockchain technology for everyday tasks such as EV charging through private and semi-private charging infrastructure. The results of a multi-actor implementation case study in Switzerland demonstrate the feasibility of the proposed blockchain framework and highlight its potential to reduce costs in a typical EV charging business model. Moreover, the study shows that the suggested framework can speed up the charging and billing processes for EV users, simplify the access to energy markets for charging station owners, and facilitate the interaction between the two through specifically designed mobile and web applications. The implementation presented in this paper can be used as a guideline for future blockchain applications for EV charging and other smart grid projects.

Keywords: blockchain; electric vehicles; Ethereum; electric vehicle charging; smart community



Citation: Dorokhova, M.; Vianin, J.; Alder, J.-M.; Ballif, C.; Wyrsh, N.; Wannier, D. A Blockchain-Supported Framework for Charging Management of Electric Vehicles. *Energies* **2021**, *14*, 7144. <https://doi.org/10.3390/en14217144>

Academic Editors: Muhammad Aziz and Antonio Cano-Ortega

Received: 21 August 2021
Accepted: 27 October 2021
Published: 1 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Background and Motivation

Profound changes are disrupting the energy industry, bringing new challenges for utilities, system operators, users, and governments around the world. The three Ds, namely decarbonization, decentralization, and digitalization, are the main drivers of a disruptive transformation in the energy sector [1]. International and national policies facilitate decarbonization by imposing ambitious targets on emissions reduction to address climate change. Moreover, the widespread adoption of renewables and improvements in energy efficiency related to power generation, transport, and usage contribute to shifting away from fossil fuels. The transportation sector accounts for 27% of global greenhouse gas emissions in the EU, 72% of which are devoted to road transport [2]. Therefore, a shift toward electric mobility is regarded as an effective way to reduce local pollution and alleviate the current energy crisis.

This decentralization is promoted by the change in the power production and consumption paradigm, where a more distributed generation is used to satisfy energy needs. The concept is tightly linked to decarbonization as the most distributed generators, such as photovoltaic (PV) or wind, are renewable in nature. However, balancing intermittent renewable resources and integrating them in the complex aging infrastructure of electric grids is not trivial and requires coordination to achieve the power supply's security and efficiency. The leading role in supporting decentralization belongs to energy storage systems (ESS) that exhibit significant cost reduction potential. From 2010 to 2020, battery costs have decreased by nearly 90%, from 1100 \$ to 137 \$ per kWh and are predicted to go down even further to 58 \$/kWh by 2030 [3]. Moreover, the global electric vehicle (EV) stock is projected to reach 140 million by 2030, which would correspond to a 7% share of the global vehicle fleet [4]. Thus, green mobility is expected to add flexibility to energy systems by participating in demand-response services enabled through the vehicle-to-grid (V2G) concept. With the capabilities of V2G, the car is transformed into a mobile battery that can be intelligently controlled to discharge electricity back to the grid, thus increasing the grid's efficiency, stability, and reliability. In a decentralized and interconnected system, such energy exchanges between the EV and the grid can be securely handled by blockchain technology, thus providing a cost-optimal and scalable solution to sustain a growing number of V2G transactions as demonstrated in [5].

The energy industry's digitalization is driven by the rapid development of artificial intelligence and advances in information and communication technologies characterized by an increased number and volume of data exchanges. The electric grid's ever-growing complexity requires new management approaches and new business models to empower efficient interactions between consumers, prosumers, grid operators, energy providers, and legal authorities. The simultaneous influence of the three Ds calls for disruptive approaches to govern energy exchanges within smart energy communities while adhering to specific requirements: decentralization, conflict-resolution mechanism, non-repudiation, efficient data management, privacy protection, and scalability [6]. Blockchain is an emerging technology that can support the transition of the energy sector into the digital era by providing secure, fast, transparent, decentralized, efficient, and low-cost operational solutions [7] in a trustless environment. The word trustless in this case signifies that neither reciprocal trust between participants nor the central authority is needed for the environment to function. Thus, blockchain is a well-suited tool to achieve key objectives of smart energy systems—cost reduction by optimizing energy processes, improved energy security, and increased sustainability—by facilitating the integration of renewables [8]. The most prominent use cases of blockchain in energy include decentralized control of power grids and demand-side management, peer-to-peer energy trading, EV management, carbon emission trading, and green certificates [9].

1.2. Blockchain

This section presents the basics of blockchain technology and provides a brief overview of smart contracts. Blockchain is a decentralized digitally distributed ledger that stores transactions of a trustless network of peers in a secure manner. Participants in the blockchain are represented as nodes that cooperate to maintain the data stored on the blockchain. The transactions are packaged in blocks of a certain size that are chained in a chronological order using advanced cryptography, thus making the blockchain grow as time progresses. The representation of a blockchain is depicted in Figure 1.

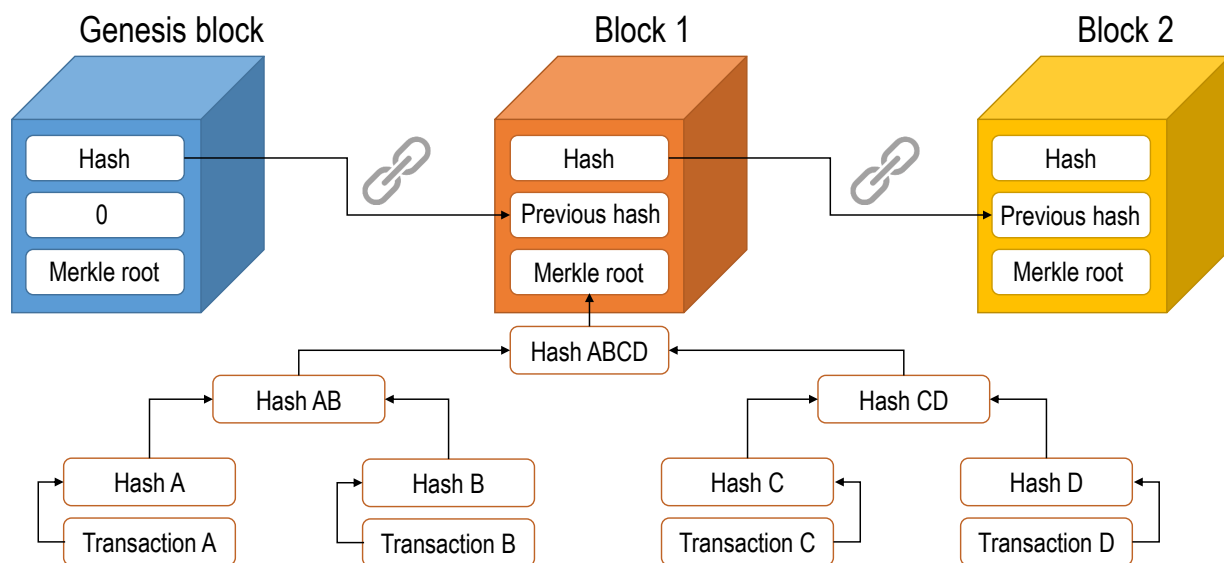


Figure 1. Blockchain concept.

A combination of hashing and encryption is used for securing the various elements of the blockchain. The hash function transforms each input data point into a unique value of a fixed length so that no one can derive the original content from the hash value. Therefore, the additional security level is added, and the size of the incoming data is reduced. The transactions packaged into the block are hashed pairwise using the Merkle tree until a single hash value remains that forms the block's Merkle root hash. The hash of the block itself is generated by hashing together the timestamp, hash of a previous block, Merkle root, and other important information. Therefore, blocks in a blockchain are interlinked as every block points to the previous block's hash value except the genesis block. Such a link structure makes blockchain immutable, as changing a single data point would entail recalculation of the hash value of the respective block. As a result, all subsequent blocks will become invalid, and the modification will be rejected.

Blockchain uses asymmetric cryptography to encrypt and decrypt data based on public–private key pairs, as seen in Figure 2. The public and private keys are cryptographically linked so that the public key can be derived from the private key using a hash function, while the opposite is impossible. The transaction begins with sender and recipient exchanging their public keys, which are essentially their addresses within a blockchain network. Afterward, the sender signs the transaction digitally using a combination of their private key and the transaction's hash value and encrypts the message with the recipient's public key. The latter verifies the sender's identity using a previously received public key and decrypts the message using their own private key.

Depending on the permission model, blockchain technology can be classified into three main types: public, consortium, and private blockchain [6]. A public or permissionless blockchain is an open network accessible to any party, where any node can view, read, and write data on the blockchain. Being essentially decentralized, public blockchain does not have any association with third parties that act as governing authorities. A consortium or permissioned blockchain is a semi-private network, where a group of organizations controls the ability of a node to join, read, or write to the blockchain. In particular cases, nodes external to the consortium can access the content without modification rights to achieve greater transparency. A private blockchain is a restricted type of a permissioned blockchain, where a single organization has full control over the network. Therefore, it offers only partial decentralization as a single entity determines participating nodes and governs the consensus process in the network of trusted parties.

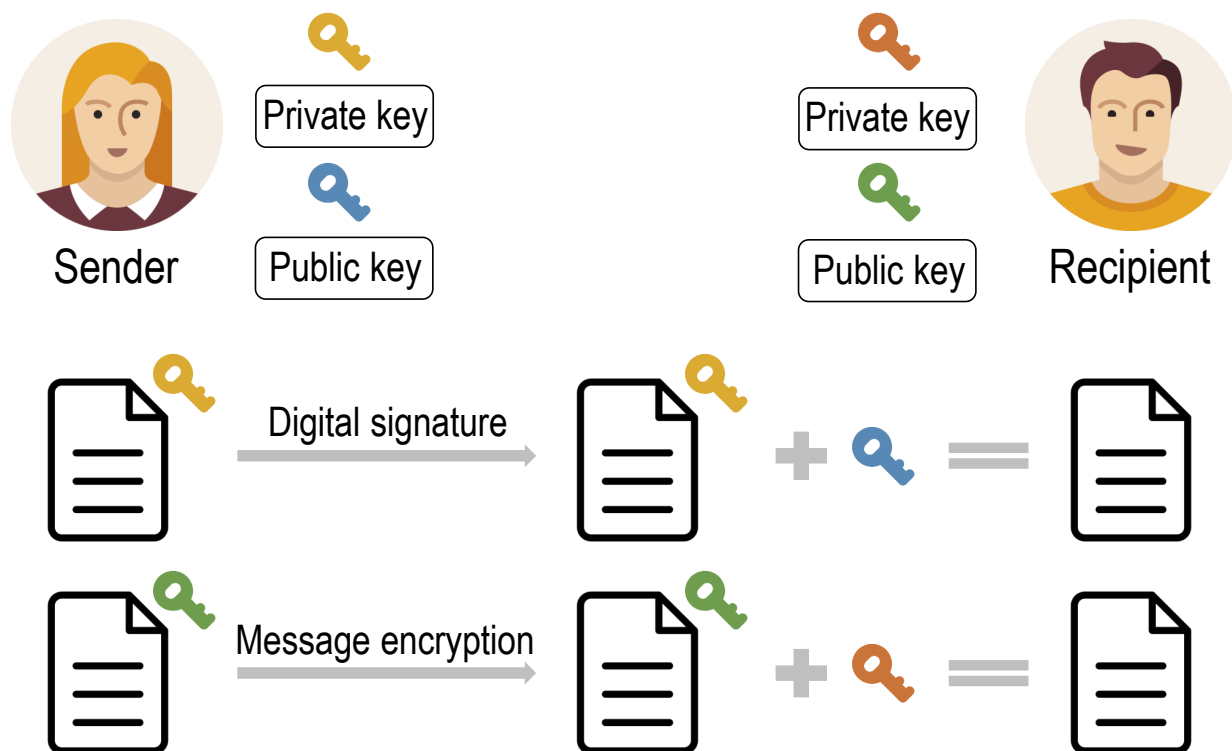


Figure 2. Message encryption and digital signature concepts in blockchain.

A blockchain node is a critical element of the infrastructure classified as either a normal or a mining node. Although all nodes are responsible for safeguarding the integrity and reliability of the blockchain, their roles vary. Normal nodes can be further classified into full and light nodes. Full nodes are data-heavy as each of them stores a copy of an entire blockchain, including transactions, timestamps, and all created blocks. Their main role is to verify, authenticate, and store the blocks in the network. Light nodes do not hold copies of the blockchain as they store only headers of each block, thus depending on full nodes to access complete validated information. Mining nodes are responsible for generating new blocks, broadcasting them to other nodes in the network, and adding them to the blockchain on top of pre-existing nodes once the validation from full nodes is received. As the blockchain technology is decentralized, a consensus mechanism is required to achieve agreement between nodes in the current state of the network. The consensus protocol's main requirements are being fault-tolerant, energy-efficient, secure, synchronized, low latency, deterministic, resilient to faulty nodes and message delays, and resistant to sophisticated hardware [7]. The most popular consensus mechanisms for blockchain are proof-of-work (PoW), proof-of-stake (PoS), proof-of-authority (PoA), practical byzantine fault tolerance (PBFT), proof-of-capacity, proof-of-burn (PoB), and proof-of-luck. In PoW, miners compete in solving complex mathematical puzzles to claim the right to add the block to the blockchain and obtain rewards. Although this process is computationally heavy and energy-intensive, it gives robustness to PoW against malicious attacks as it is almost impossible to acquire 51% of computing power to gain control of the network. The PoS and PoB are examples of more environmentally friendly consensus mechanisms that do not require energy-hungry calculations. The prior algorithm favors miners with larger amounts of cryptocurrency at stake, while the latter demands miners to invest coins to an unspendable address.

The advent of the Bitcoin blockchain [10] in 2009 has transformed the concept of digital payments by removing the middle-man and introducing cryptocurrency. Since then, the scope of blockchain expanded far beyond financial applications when the Ethereum-decentralized platform [11] established smart contracts previously proposed in [12]. The

user-created smart contract is essentially an executable piece of code that resides on the blockchain and runs automatically when predefined initial conditions are met. To make the smart contract functional, one has to define the parties that enter into the agreement, the agreement's subject, and specific terms of the contract. The latter include requirements expected from all participants, mathematical rules that define the contract's enforcement, and rewards and penalties associated with the smart contract's successful run. To execute the smart contract, the operations comprising it are performed by miners within the Ethereum Virtual Machine. Each operation, such as addition or subtraction, is associated with a specific gas cost that measures the computational effort required to carry it. Thus, the total amount of gas needed by the contract depends largely on its size and complexity. The resulting transaction fee can be calculated by multiplying the gas by its unit price, which is determined based on the supply and demand of computing power.

The main advantages of smart contracts are their speed, accuracy, and trust, as all nodes in the blockchain witness the execution of a smart contract, savings due to third-party elimination, and security due to the cryptographic basics of blockchain. The disadvantages are uncertain legal status and high implementation costs, as an experienced programmer with expertise in the field is required to establish the agreement's rules correctly. Moreover, the deployment of a smart contract on the blockchain is irreversible, meaning that it cannot be altered once created. Although immutability is the main asset of smart contracts, encoded errors and loopholes can threaten security and lead to hacker attacking, which happened with Decentralized Autonomous Organization (DAO) [13], resulting in a radical change of Ethereum's blockchain protocol, referred to as a hard fork.

1.3. Literature Review

The use of blockchain in smart energy systems has been a topic of growing research interest over the past few years. The unique features of blockchain, together with specific challenges of the energy sector, motivate researchers to explore new implementation scenarios, define various objectives, and examine diverse energy systems. Previous works, particularly in the EV management application, can be grouped into three categories according to the type of blockchain deployed: public, consortium, and private blockchain.

The first group of studies uses public blockchain, mainly Ethereum, as a transaction mediator in various energy management tasks within smart-energy communities. Researchers in [14] developed an autonomous charging station selection process using the smart contract capability of the Ethereum network. The design aims to reduce EV charging costs while considering the planned routes, traffic conditions, user preferences, and additional incentives from the energy provider. Similarly, the authors in [15] deployed a charging station scheduling algorithm based on the Bitcoin lightning network to minimize various user-incurred costs, waiting time, and distance traveled to the charging point. The work in [16] resolved the optimal charging station selection problem using the cost–distance trade-off. Although the solution was not implemented practically, both Bitcoin and Ethereum blockchain networks were discussed as a reasonable choice for a small number of EVs. A real-time cryptocurrency-based incentive approach was proposed in [17] to maximize the usage of renewables in the energy system consisting of EV, charging station, PV generation, and battery ESS. Researchers in [18] suggested coupling the Ethereum blockchain with an EV valley-filling charging strategy to reduce overhead on the electric grid. Another example of Ethereum usage was demonstrated in [19], where the energy flows of a microgrid consisting of EVs with their charging stations, residential homes, and renewable generation were managed to minimize the impact of injecting or consuming an excessive amount of power into or from the grid. The authors of [20] developed an Ethereum-based EV charging coordination scheme by substituting the standard PoW consensus mechanism with the PoA.

The second group of work deploys consortium blockchain. Researchers in [21] applied an iterative double-auction mechanism to resolve the optimal energy allocation problem. Their system included plug-in hybrid EVs, charging stations, local energy aggregators,

and smart meters. The solution aimed to maximize social welfare using consortium blockchain with the PoW consensus mechanism. The authors in [22] focused on satisfying the needs of EVs while maximizing the operator's utility in the smart community with PV generation and ESS. The formulated energy blockchain system deployed a reputation-based delegated byzantine fault tolerance algorithm to reach consensus among the participating nodes. Yet another consensus scheme, proof-of-reputation, was implemented in [23] to maintain regional energy balance and maximize EV user's utility and satisfaction whilst deploying wireless EV charging technology. The capabilities of EV charging were extended toward V2G and vehicle-to-vehicle concepts in the work of [24], where the optimal charging scheduling was considered to maximize EV satisfaction while minimizing the charging costs. Researchers in [25] gathered EV manufacturers, EV service providers, EVs, charging station operators, and software providers in a consortium blockchain enabled through Hyperledger Fabric with an improved PBFT consensus mechanism. Their work aimed to minimize overall load variance under the distribution network by shifting peak loads while respecting power flow constraints and EV charging demands. The authors of [26] included the government as a participant of the consortium blockchain to achieve fair and reasonable EV allocation to charging stations. The proposed EV management scheme addressed the profit allocation problem among various energy companies that provide EV charging services. An incentive economic-based mechanism enabled through EV coins was suggested in [27] with a particular focus on PV generation. The authors developed a prioritization ranking algorithm to guide the EV charging patterns toward maximizing renewables' utilization. Researchers in [28] considered the internet of EVs and local energy aggregators for the sake of facilitating demand-response measures through consortium blockchain with a standard PoW consensus mechanism.

The third group consists of a few studies that implemented private blockchain for energy exchange purposes. The authors in [29] deployed a private Ethereum blockchain with PoA consensus mechanism among the network of four entities: EV, energy provider, smart meter, and utility company. The blockchain in their system manages the market auction mechanism and the billing procedure to maximize social welfare. Researchers in [30] deployed a practical BFT consensus mechanism in a private blockchain to minimize operation, transportation, and transaction costs in a peer-to-peer trading network of loads and EVs designated as prosumers.

1.4. Contribution

As was demonstrated in the literature review, very few studies go beyond theoretical framing of blockchain for EV management towards practical implementations in the real world. Therefore, motivated by the developments mentioned earlier, in this paper, we aim to illustrate a concrete test case implementation of a blockchain-based EV charging framework, tightly interlinked with physical infrastructure in a trustless environment. Our particular contributions include the following points:

- We provide an extensive comparison between some of the most popular blockchain platforms, such as AragonOS, Energy Web Chain, Hyperledger Fabric, and Ethereum, across a set of comprehensive criteria, such as deployment, maintainability, and scalability, crucial for the proof of concept.
- We develop the first Ethereum-based architecture of the EV charging management framework, tightly interlinked with real-world infrastructure. Using a Solidity programming language, we design a particular smart contract that guides the EV charging process while ensuring the correct accounting for participating entities.
- We build a web interface and a mobile application based on the client's journey to provide a user-friendly experience of EV charging encompassing the capabilities of the Ethereum blockchain. The two created instruments serve as the media for EV and charging station owners to monitor the charging process while being securely credited for respective energy flows.

- We demonstrate the application of the proposed blockchain-enabled EV charging framework on a real-world case study and document the whole EV charging process. Moreover, we assess our Ethereum-based framework's performance by evaluating some of the most common metrics in the field, such as transaction latency and fees.

The remainder of this paper is structured as follows. Section 2 presents the research approach, particularly elaborating on the choice of blockchain, system architecture, and smart contract design. Section 3 discusses the system's implementation details and introduces the web interface and mobile application. Section 4 presents the real-world case study used to validate the approach and summarizes the main results. Section 5 concludes the paper.

2. Methodology

2.1. Choice of Blockchain Technology

Despite being a powerful and flexible technology, blockchain is not a uniform solution to any problem. Therefore, one has to choose the most suitable blockchain for the problem at hand and customize it if necessary. According to [31], the following blockchain features have to be considered when choosing a reasonable blockchain implementation: consensus mechanism, speed, permission model, and resilience. The latter signifies the capability of a blockchain-based system to resist to attacks and malicious behaviors. In the current work, we aim to develop a blockchain-enabled energy exchanges framework applicable to real-world scenarios. Therefore, we add the following criteria to the choice of appropriate blockchain implementation:

- **Deployment** follows the blockchain life-cycle from the development, including listing the requirements and actual programming, to the final release of the system into production.
- **Maintainability** refers to the degree of difficulty and effectiveness with which the intended maintainers can modify the blockchain system through updates. The modifications can contain corrections and error handling, system improvements, and adaptation.
- **Scalability** signifies the capability of blockchain to handle the growing amount of participants and transactions. In particular, it is expressed in how fast the blockchain can reach the consensus among nodes and add a new transaction into a block, and how many transactions per second it can process.

In this work, we analyzed and compared four popular development frameworks for decentralized applications (DApps) that support blockchain implementation of smart contracts: AragonOS [32], Hyperledger Fabric [33], Energy Web Chain (EWC) [34], and Ethereum basic smart contract [11]. Table 1 summarizes the main features used for qualitative comparison among considered frameworks, where the speed is measured in transactions per second (TPS). One has to note that despite planning to switch to Aragon Chain with the PoS consensus mechanism in the upcoming future [35], the AragonOS currently deploys DApps on Ethereum's main network. Therefore, we directly compare the permissioned Hyperledger Fabric with permissionless Ethereum and EWC.

Table 1. Qualitative comparison of various blockchain frameworks.

| Blockchain | Consensus | Speed | Permission |
|--------------------|------------------|----------|----------------|
| Hyperledger Fabric | voting-based | 3000 TPS | permissioned |
| Ethereum | lottery-based | 15 TPS | permissionless |
| EWC | reputation-based | 76 TPS | permissionless |

As can be seen in Table 1, Hyperledger Fabric and Ethereum use different types of consensus algorithms, which are Apache Kafka and PoW, respectively. The lottery-based Ethereum consensus mechanism scales well to a large number of nodes but results in a longer time to finality than Hyperledger Fabric. Finality signifies the state of the blockchain under which the transaction cannot be canceled, reversed, or changed by any

of the network's participants under any circumstances. If two winning miners simultaneously propose a new valid block, the blockchain will experience a temporary fork, and the acceptance of the blockchain state by all nodes will be delayed. The voting-based algorithm of Hyperledger Fabric, instead, provides low-latency finality but scales less efficiently as the time to reach consensus increases with the expansion of the network. Moreover, the algorithm's crash-fault-tolerant nature prevents the blockchain from achieving agreement in the presence of faulty or malicious nodes [36]. The PoA reputation-based consensus mechanism of EWC is a hybrid approach that resides in between the lottery-based and voting-based consensus mechanisms. The PoA relies on a set of trusted miners, called authorities, to take on the leader's responsibility for a new block creation in a rotation manner. Despite being faster and more energy-efficient, the PoA consensus mechanism does not claim a full decentralization while questioning immutability. As authorities' identities are visible to everyone in the network, such things as censorship, blacklisting, and third-party manipulation can be potentially achieved, thus compromising the safety of the blockchain. In energy-specific Ethereum-based EWC, the largest global energy companies host the validation nodes, thus executing the power to approve the new blocks in a highly regulated energy market. The speed of considered blockchain frameworks varies significantly, as Ethereum currently processes only 15 TPS, while Hyperledger processes 3000 TPS. However, researchers in [37] recently demonstrated an upscale of Hyperledger Fabric to 20K TPS, and Ethereum 2.0 is expected to yield 100K TPS in the upcoming future with the switch to PoS consensus mechanism. The EWC is currently capable of processing around 76 TPS [38]. However, the Energy Web Foundation organization that launched the EWC repeatedly claims that the TPS metric is not suitable to assess the scalability of the blockchain correctly.

To discuss the resilience of considered blockchain frameworks, one has to identify potential cyber threats. The attacks on the blockchain can be grouped into five main categories: blockchain network attacks, user wallet attacks, smart contract attacks, transaction verification mechanism attacks, and mining pool attacks [39]. The permissioned nature of Hyperledger Fabric adds a layer of security by authorizing access to only a predefined pool of participants. Moreover, the business purpose design of Hyperledger Fabric requires the system to quickly recover from attacks without compromising sensitive client data. Researchers in [40] have concluded that blockchains using different programming languages and architectures have different vulnerabilities and are thus susceptible to different types of attacks. The smart contracts of Ethereum written in Solidity language are considered to be more vulnerable than the chaincodes of Hyperledger Fabric programmed in Go. The EWC enterprise-grade blockchain exhibits good resilience, specifically due to the known list of validators that contribute to the overall integrity and security of the network. However, the underlying PoA consensus mechanism is widely criticized for being susceptible to distributed denial-of-service attacks in the case of insufficiently large pool of validators.

Table 2 summarizes our assessment of blockchain systems' deployment, maintainability, and scalability on a scale from 1 to 5, where 5 signifies easiness and 1 means difficulty. Notably, the criteria considered do not have the same weight, with deployment and scalability being the most and the least important, respectively, for our concept of real-world implementation.

Table 2. Assessment comparison of various blockchain frameworks.

| Blockchain | Deployment | Maintainability | Scalability | Total |
|--------------------|------------|-----------------|-------------|-------|
| Weights | 0.5 | 0.3 | 0.2 | |
| AragonOS | 2 | 5 | 3 | 3.1 |
| Hyperledger Fabric | 3 | 3 | 4 | 3.2 |
| Ethereum | 5 | 2 | 2 | 3.5 |
| EWC | 4 | 3 | 3 | 3.5 |

We consider the deployment of the basic smart contract in Ethereum to be more straightforward than the deployment in Hyperledger Fabric and AragonOS, despite the latter two providing smart contract templates for cloning. Indeed, it consists of two steps successive to writing the contract code itself: compiling the smart contract into bytecode and deploying it by sending an Ethereum transaction without specifying any recipients. All these actions can be performed through Remix IDE. In Hyperledger Fabric, the multi-layered access control framework complicates the process as one has to first install the smart contract on peers and then instantiate it on the channel. However, a single chaincode can define several smart contracts at once, which simplifies the development. The AragonOS has a multi-step deployment procedure with the installation of Node.js runtime environment, Web3 provider to interact with Ethereum, MetaMask to sign transactions, and Aragon command line interface to install the new app. The Ethereum-based EWC offers a comprehensive toolkit with open-source templates of energy-specific digital applications to speed up the development of customized DApps. In addition, the general procedure for deploying a smart contract is similar to the one of Ethereum, where installing the mandatory packages and related development environments is required. However, a preliminary step of setting up the Energy Web Decentralized Operating System is necessary. Moreover, a recent analysis of the EWC network has shown that over 80% of the smart contracts were deployed from only three participating entities [41], thus indicating either the lack of interest from the general public, the novelty of EWC, or the difficulty to deploy the smart contracts in the energy field.

The AragonOS shows the highest maintainability thanks to easily updating the smart contract to a newer version. Such a feature is available in AragonOS due to the specific design solution: the smart contract's logic is decoupled from its location using proxies. Therefore, developers can fix bugs and push enhancements without changing the address of the smart contract. Similarly, in Hyperledger Fabric, the contracts can be upgraded; however, one has to install the contract with the same name and a different version on all peers before upgrading the smart contract. If the order is reversed, certain peers will lose their ability to participate in the network by endorsing transactions. The smart contracts on Ethereum have the lowest maintainability as they are immutable by default. However, certain approaches to enable upgradability exist. To release the smart contract update, one can deploy a proxy contract that delegates the execution of methods and functions to implement smart contract. Such a methodology allows switching the logic contract easily, as users interact only with a proxy contract. However, one has to think of such a maintenance option, while the smart contract is still in the design stage and is not deployed on the blockchain. The life-cycle of smart contracts on EWC network is supposedly guided by the OpenZeppelin secure smart contract library that gives a possibility to securely destroy and pause the smart contract. However, we did not find any comprehensive description of its functionality on the EWC blockchain.

The scalability criterion was previously discussed with the qualitative comparison of blockchain frameworks. The Hyperledger Fabric currently scales better than AragonOS, EWC, and Ethereum due to its permissioned nature and absence of the PoW consensus mechanism. However, upcoming releases of Ethereum 2.0 that will potentially include sharing to increase the amount of TPS, and Aragon Chain aims to resolve the scalability issue. For EWC blockchain, the great scalability promises are held due to its PoA consensus mechanism. On a side note, the scalability of AragonOS is seen as less of a challenge due to focusing mainly on governance and not on transactions. In particular, large decentralized organizations can define a quorum to simplify the consensus process, thus eliminating the need to achieve 51% majority of the whole network.

To summarize, the four considered blockchain frameworks for smart contract development vary in their purpose, features, and challenges they are facing. As EWC and Ethereum both acquired the same amount of points in our subjective evaluation scheme, we had to make a choice between the two. Despite the prior being suitable for energy-related applications specifically, its consideration of not being fully decentralized has contributed

to our decision. Thus, with the focus on fast deployment and high accessibility of the blockchain network for testing purposes in real-world scenarios, we concluded that the choice of the basic Ethereum smart contract is the most appropriate.

2.2. Blockchain System Architecture

Figure 3 presents the system's architecture that includes the main hardware assets and respective interaction flows between them. It should be noted that the demonstrated system architecture is scalable and can accommodate several hotels, electric vehicles, and charging stations, despite the fact that they are drawn in single quantities. The information flows are shown using the dashed line, while the power flows are depicted with the solid line. The following elements compose the architecture of the blockchain system:

- The **PV installation** in our system belongs to the hotel and is usually located on the rooftop of the building. The renewable power from PV is used to satisfy the load demand of the hotel. The excess of the PV production is supplied to the charging station when needed or is fed back to the utility grid by the hotel.
- The **utility grid** provides power to the hotel and the charging station whenever there is demand.
- The **hotel** is the main physical entity in the system architecture. The hotel is characterized by its load demand, which can be satisfied either by the rooftop PV or by the utility grid. The hotel sends the smart meter power measurements and the information about PV production to Time Series Database (TSDB). The hotel owner or the hotel manager interacts with the Server to issue charging requests to the charging station in manual mode.
- The **charging station** itself is not endowed with the layer of intelligence. Therefore all interactions with the charging station are conducted through the UniPi controller. The charging station is either AC or DC and can be operated in both manual and automatic modes. In manual mode, the maximum charging current is set by the hotel owner, while in automatic mode, the current is regulated according to the optimized EV charging strategy. The charging process is supported by both the utility grid and the PV installation. Once the charging is complete, the charging station returns to UniPi the charging status and the amount of energy consumed in kWh.
- The **UniPi** is a programmable logic controller mounted inside the charging station. The UniPi enables the automatic control of the charging process through the commands received from the Server. The internal API allows the UniPi to send requests to the charging station using Modbus in write and read modes. Particularly, the UniPi can set the charging station's status, the energy consumption required, and the maximum amount of amps the charger can deliver to the EV. Once the charging is complete, the UniPi returns the overall amount of energy consumed during the charging process in kWh to the Server.
- The **EV** on the scheme represents both the vehicle itself and the EV driver, who interacts with the UniPi of the charging station using a mobile application. When the EV arrives at the charging station, the driver optionally sends out the information about the EV's current state of charge and the amount of time the EV can spend at the charging station until the next departure. This information is used to optimize the charging process and deliver the highest PV self-consumption possible while satisfying the EV's charging needs and maximizing the state of charge at departure. Further information about the optimization procedure can be found in [42]. If no supplementary information is provided by the driver at arrival, the charging process proceeds without the optimization feature.
- The **TSDB** stores all data collected from the hotel. Besides the load consumption and PV production, the information might contain hot water usage and other measurements related to the hotel's equipment, such as heat pumps, boilers, energy storage, etc. TSDB feeds the data to the Server for visualization purposes in the front-end and for determining the optimal EV charging strategy.

- The **Meta Database** contains the hotel-related data used for creating the front-end of the visualization dashboard. The information includes the hotel's profile, the list and the characteristics of the installed equipment, etc. When the system scales up to include multiple hotels, Meta Database becomes particularly useful for differentiating the static hotel-related data from dynamic time-series data stored in TSDB.
- The **Server** links the elements of the system architecture and enables communication from within. Moreover, it provides the visual dashboard to the hotel owner, where the latter can view the information in the interactive mode and issue the requests to the charging station. The data displayed include the hotel's measurements and the information related to the concluded charging processes such as charging start and finish times, duration, cost, and energy content. In particular, the hotel owner can browse through the charging history of either the charging station or the EV that belongs to the hotel.

The following section describes how the blockchain facilitates the energy exchanges within the system using Ethereum smart contracts.

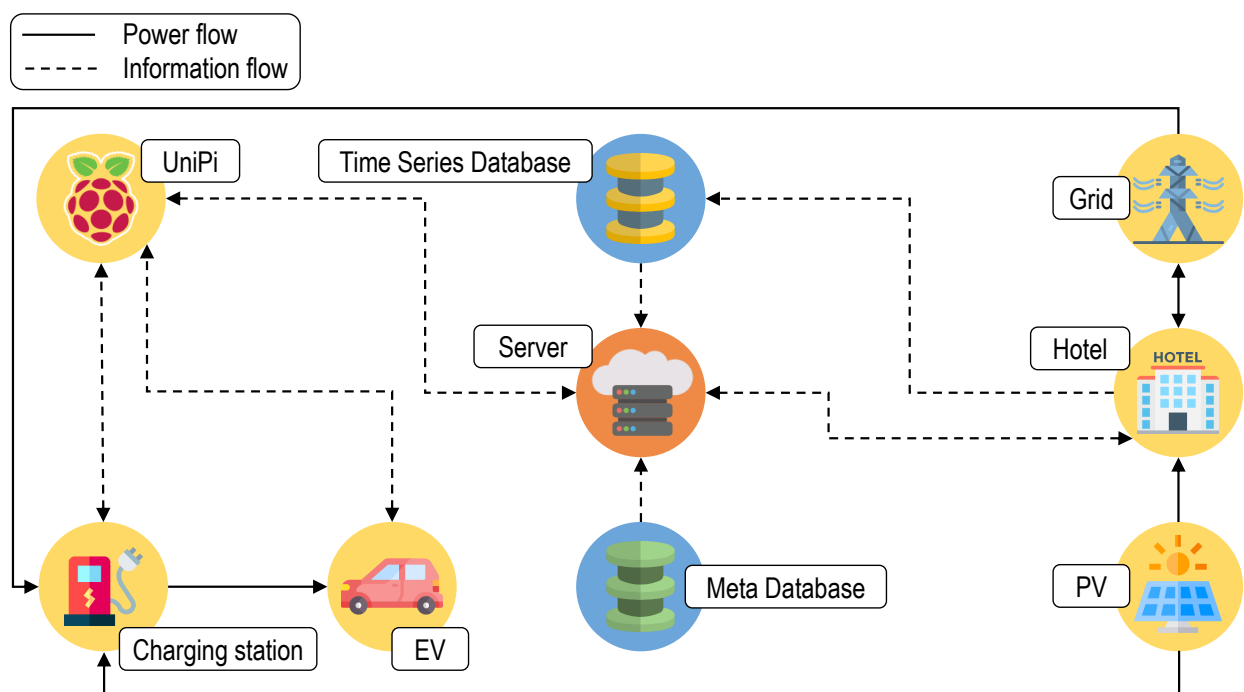


Figure 3. Blockchain system architecture.

2.3. Smart Contract

The Ethereum smart contract is written in Solidity programming language and is used to settle energy exchanges between charging stations and EVs in the form of blockchain transactions. The following entities with their particular attributes are considered participants of the smart contract:

- **Hotels** are the central parties in the smart contract defined by their Ethereum addresses. Each hotel possesses at least one charging station and eventually one EV, uniquely differentiated by their `idChargingStation` and `idVehicle` identifiers. The charging stations and EVs do not have their own Ethereum addresses, as in the charging process they act on behalf of the hotel they belong to. Therefore, a charging transaction is conducted between two hotels, where one behaves as the energy provider and the other as the energy consumer. If the EV is charged at the hotel it belongs to, this hotel takes on a double role resulting in a transaction with itself. Each hotel stores the history of its charging transactions in the `chargingTransactions` list.

The hotels' respective energy balances are kept in the `hotelBalances` list, which can be queried by the hotel's address. To participate in the charging process by issuing a charging transaction, the hotel's registration is required and is verified using the `onlyRegisteredHotels` modifier.

- The **owner** of the smart contract is the sender of the transaction that deploys this smart contract on the blockchain. The owner is characterized by an Ethereum address and is endowed with two exclusive capabilities enabled by the `onlyowner` modifier: registering new hotels and resetting energy balances. The prior allows the owner to add a new hotel to the `registeredHotels` list, thus enabling its participation in energy exchanges between EVs and charging stations. The latter gives the contract owner a possibility to reset the energy balance of a specific hotel to zero if needed. Moreover, the owner has the right to disable a hotel by modifying its status in the `registeredHotels` list.

The `ChargingTransaction` object of the smart contract is defined as a custom structure containing the state variables presented in Table 3:

Table 3. The charging transaction object.

| Field Name | Data Type |
|-----------------------------------|-----------|
| <code>addressHotelSupplier</code> | address |
| <code>addressHotelConsumer</code> | address |
| <code>startDate</code> | uint |
| <code>endDate</code> | uint |
| <code>energyConsumption</code> | int |
| <code>idVehicle</code> | uint |
| <code>idChargingStation</code> | uint |

Where `addressHotelSupplier` is the address of the hotel owning the charging station, `addressHotelConsumer` is the address of the hotel owning the EV, `startDate` and `endDate` are the Unix times of the beginning and the end of the charging process, respectively, and `energyConsumption` is the amount of energy transmitted in the process.

At the beginning of the smart contract, we define the following public and private mapping functions, where the prior type enables an automated getter-creation in the Solidity language:

- `mapping (address => bool) public registeredHotels` allows for a quick verification of the hotel's registration.
- `mapping (address => int) public hotelBalances` gives a possibility to check the balance of the hotel of interest.
- `mapping (address => ChargingTransaction[]) private chargingTransactions` associates the list of charging transactions with the hotel's address and allows viewing the charging history from within the smart contract.

The `addChargingTransaction` function is the core of the smart contract. The function uses the `onlyRegisteredHotels` modifier, thus allowing only the charging stations from registered hotels to initiate charging transactions. The input to the function contains the following variables inherited from the `ChargingTransaction` object:

- `address _consumerAddress`
- `uint _startDate`
- `uint _endDate`
- `int _minusEnergyConsumption`
- `int _energyConsumption`
- `uint _idVehicle`
- `uint _idChargingStation`

It is important to notice that the supplier's hotel address is absent from the function input as it is automatically defined by the `msg.sender` property of the charging station issuing the transaction. The `_minusEnergyConsumption` variable, which differs from the `_energyConsumption` only by its negative sign, is introduced to avoid the additional cost on blockchain related to the subtraction of the `_energyConsumption` value.

The function's body contains the following expressions:

- `chargingTransactions[msg.sender].push()` is used to record the charging transaction on the side of the hotel owning the charging station. The push function appends the `ChargingTransaction` with the amount `_minusEnergyConsumption` to the end of the transaction list.
- `chargingTransactions[_consumerAddress].push()` stores the transaction on the side of the hotel that owns the EV. The respective energy amount `_energyConsumption` is positive.
- `hotelBalances[msg.sender] -= _energyConsumption` adjusts the energy balance on the supplier's side.
- `hotelBalances[_consumerAddress] += _energyConsumption` adjusts the energy balance on the consumer's side.

The detailed smart contract class diagram is presented in Figure 4.

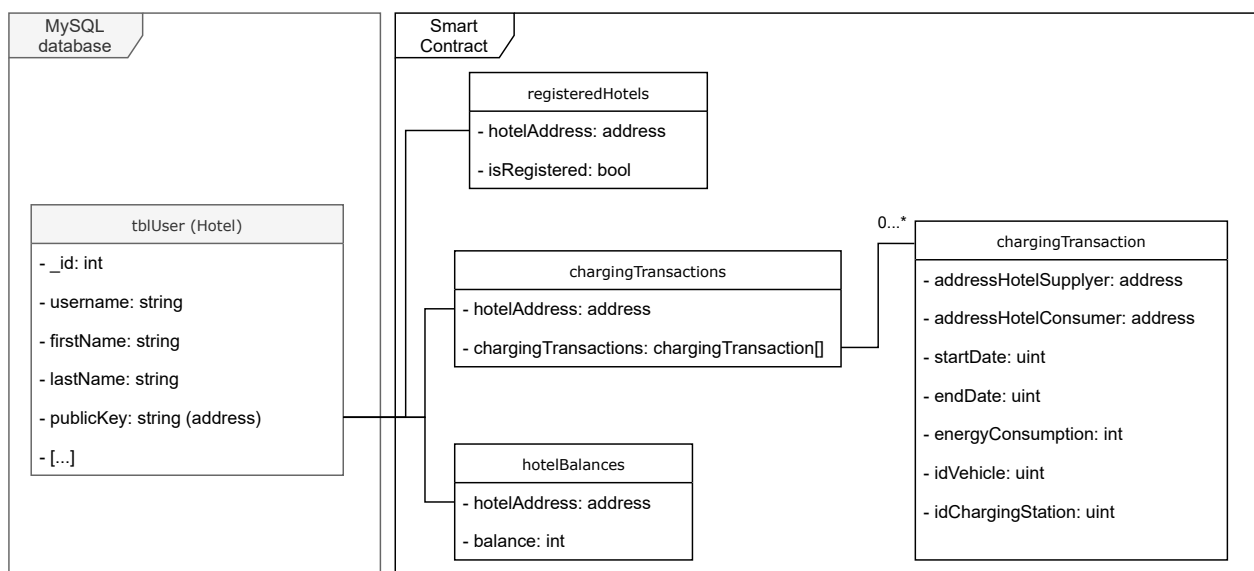


Figure 4. Smart contract class diagram.

3. Implementation

3.1. Pilot Site

The case study is based on the Digitalization project [43] conducted within the research framework of the SCCER FURIES [44]. The project, established as part of the activities related to the Swiss National Action Plan on Digitalization [45], integrates a network of EVs and charging stations available to guests staying in the hotels of the Val d'Hérens alpine region in Switzerland. Each of the eight partner hotels owns one charging station and at least one EV, allowing guests to explore the region with maximum independence and minimum harm to the environment. The EVs are rented to the hotels' guests daily free of charge, based on the principle "pay what you want".

3.2. Process Flow

Figure 5 describes the steps of the EV charging process flow. The main entities implicated in the process are the hotel owners and managers, EV charging stations, and hotel guests. The prior participate in the process using the web interface detailed in Section 3.4,

while the latter issue charging-related commands through the mobile application described in Section 3.5.

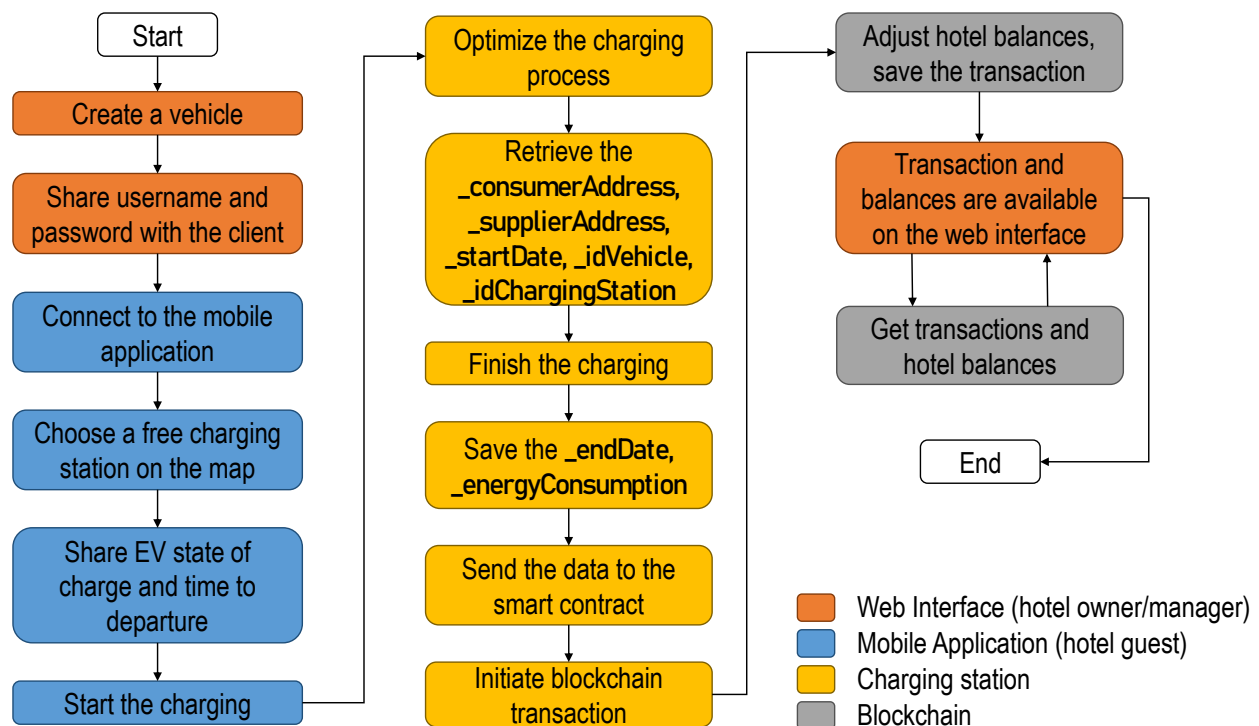


Figure 5. EV charging process flow.

To enable the EV's participation in the blockchain-supported charging process, the hotel owner must register the EV in the web interface. Once the EV can be found in the hotel's digital vehicle inventory, this step becomes unnecessary. The registration procedure generates a unique vehicle identifier `idVehicle` together with the username–password pair. The latter is shared with hotel guests to enable access to the mobile application used for managing the EV charging process. Besides the standard way to log in, the EV user can utilize the generated QR code, thus avoiding memorizing complex login details. It is expected that the password and QR code will be automatically regenerated.

Once the guest has rented the EV and has connected to the mobile app, the charging process unfolds as follows. First, the EV driver chooses an unoccupied charging station on the map. Second, the user optionally shares the EV's state of charge and the available time to spend at the charging station. If transmitted by the EV driver, these details allow the charging station to optimize the charging process according to the algorithm described in [42]. Third, the EV user starts the charging process by clicking the dedicated button on the mobile application. Otherwise, the charging process can be initiated by the hotel's manager from the web interface. From that moment on, the charging station manages the EV charging process flow.

During the EV charging, the charging station collects the data required to populate the `ChargingTransaction` object and execute the `addChargingTransaction` function. The charging station retrieves such data from the web interface using the application programming interface (API). The EV charging ends with either the user terminating the process through the mobile application or the charging station when the maximum state of charge is reached. Once the charging is complete, the charging station initiates the blockchain transaction by sending the data to the smart contract. The hotel owner can access the history of charging transactions and their details by interacting with the blockchain.

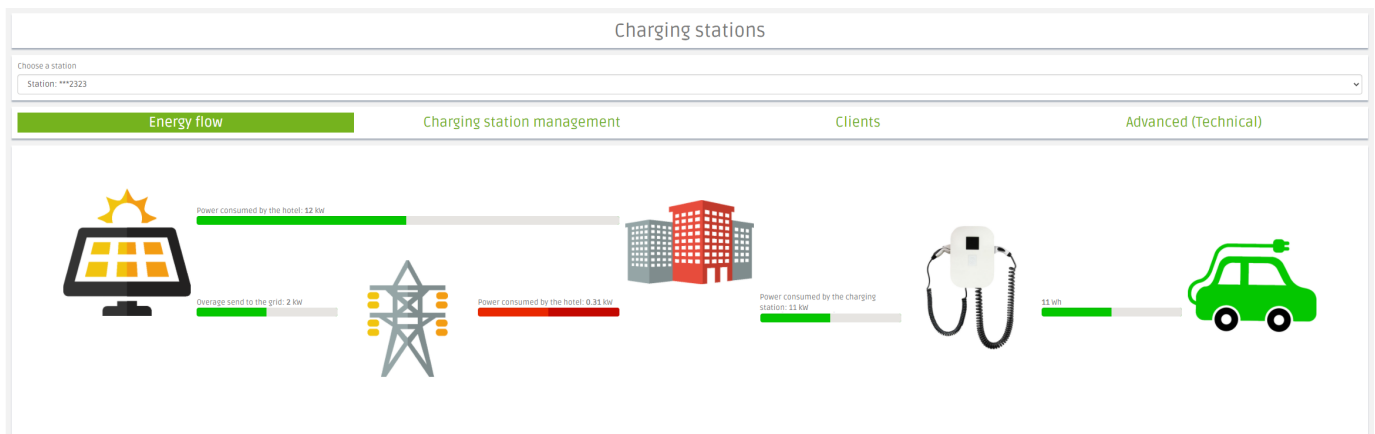


Figure 7. Energy flow tab on the web interface.

Figure 8 demonstrates the charging station's control panel, where the hotel owner can see the charging station's status and manipulate the charging process. The owner can particularly influence the charging station's availability and switch to the charging process's manual control mode. The latter gives the hotel owner the possibility to set the charging station's desired current, which translates directly to the power output.

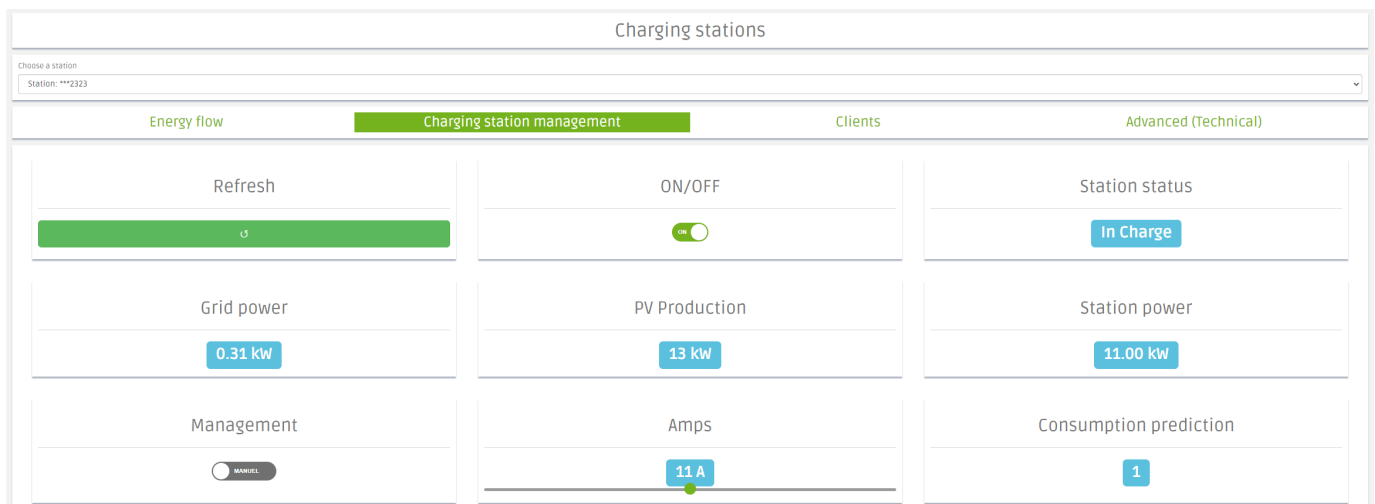


Figure 8. Charging station management tab on the web interface.

The clients' tab depicted in Figure 9 is the hotel's point of interaction with the vehicle inventory and the blockchain system. The history of charging transactions is displayed on the tab's left-hand side, where the date, identifiers of the EV and the charging station, energy, and charging duration can be seen. The hotel's energy balance is placed underneath the charging transactions and can be used for the hotel's reporting at desired time periods. The tab's right-hand side provides the hotel owner an overview of the EV inventory with the possibility to register additional vehicles.

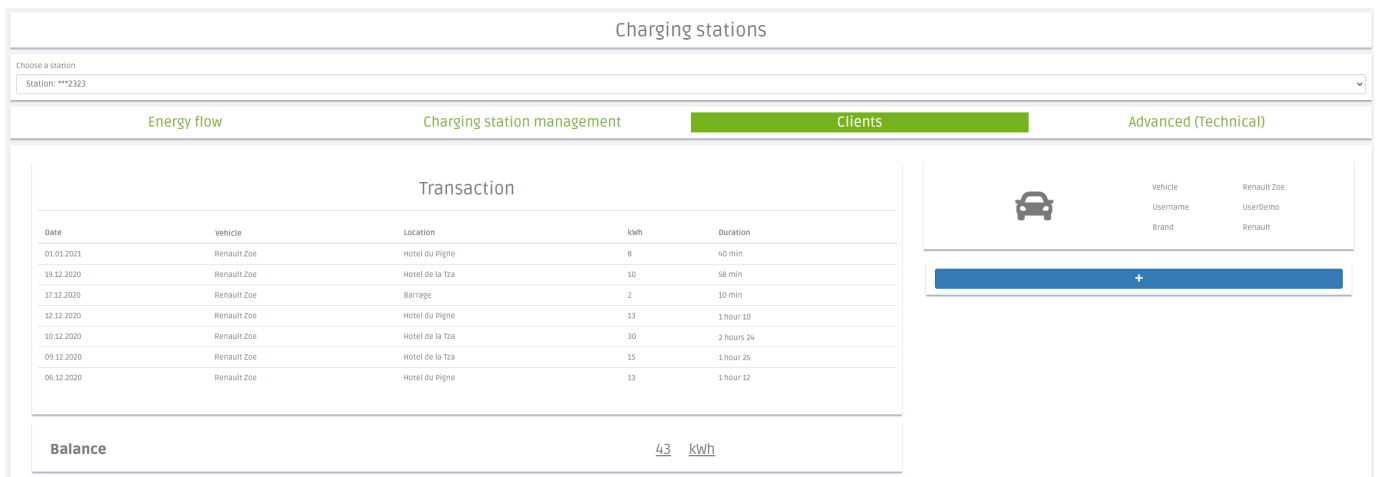


Figure 9. Clients tab on the web interface.

3.5. Mobile Application

The mobile application is the means for the EV user to interact with charging stations participating in the green mobility program. The application, developed in NativeScript, is available for both mobile operating systems Android and iOS. The following figures demonstrate the mobile application's design and functionalities.

Figure 10 depicts the login page, where the EV user is asked to either input the username–password pair or scan the QR code. Both are transmitted to the user by the hotel owner or manager at the beginning of EV rental. Figure 11 shows the EV user's profile, where the username is displayed. In the future releases of the application, other information about EV, such as model, battery capacity, charger type, etc. will appear on the account screen.

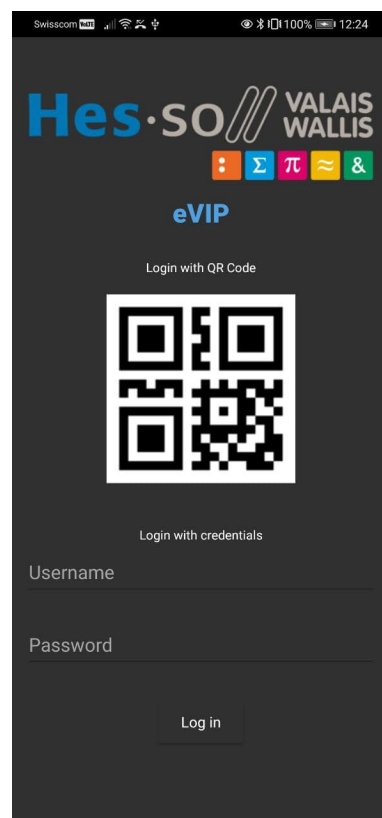


Figure 10. Login page.



Figure 11. Account page.

Figure 12 presents the map screen that depicts the charging stations in the area and their status, thus helping the EV driver choose where to charge. If the charging station is not occupied, the icon's color on the map appears green and red otherwise. Clicking on the icon opens the details about the charging station, as shown in Figure 13. In particular, the user can see the name of the charging station, its address, the hotel it belongs to, the types of charging plugs at its disposal, and the charging station's availability. Optionally, to optimize the charging process concerning the hotel's demand and PV production, the EV driver can input the EV's state of charge and the time planned to spend at the charging station. Otherwise, the required charging time is calculated based on the charging power of EV. The charging process is activated using the button.

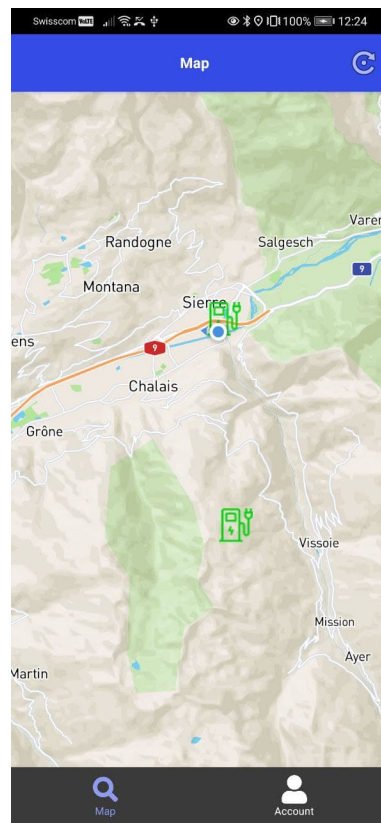


Figure 12. Map of charging stations.

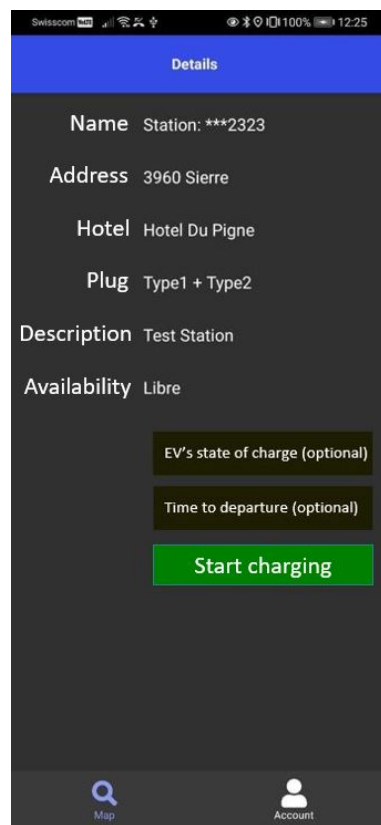


Figure 13. Charging station details.

4. Results

To validate the proposed blockchain system architecture and demonstrate its application to the use case of EV charging, we recreate the process flow depicted in Figure 5 in a real-world case study. The two hotels, Hotel du Pigne and Hotel Aiguille de la Tza, both located in Val d'Hérens region in Switzerland, act as the demonstration participants. The former owns the EV, while the latter provides the charging station. Thus, the test aims to show how the EV-charging procedure unfolds in the blockchain-enabled EV management framework. The experiment is conducted using the UniPi Axon S105 programmable logic computer installed inside the charging station, with the following characteristics: 1.2GHz quad-core ARM CPU, 1GB RAM, 8GB eMMC onboard memory, and 1Gbit Ethernet connection. Figure 14 depicts the charging infrastructure used for testing.

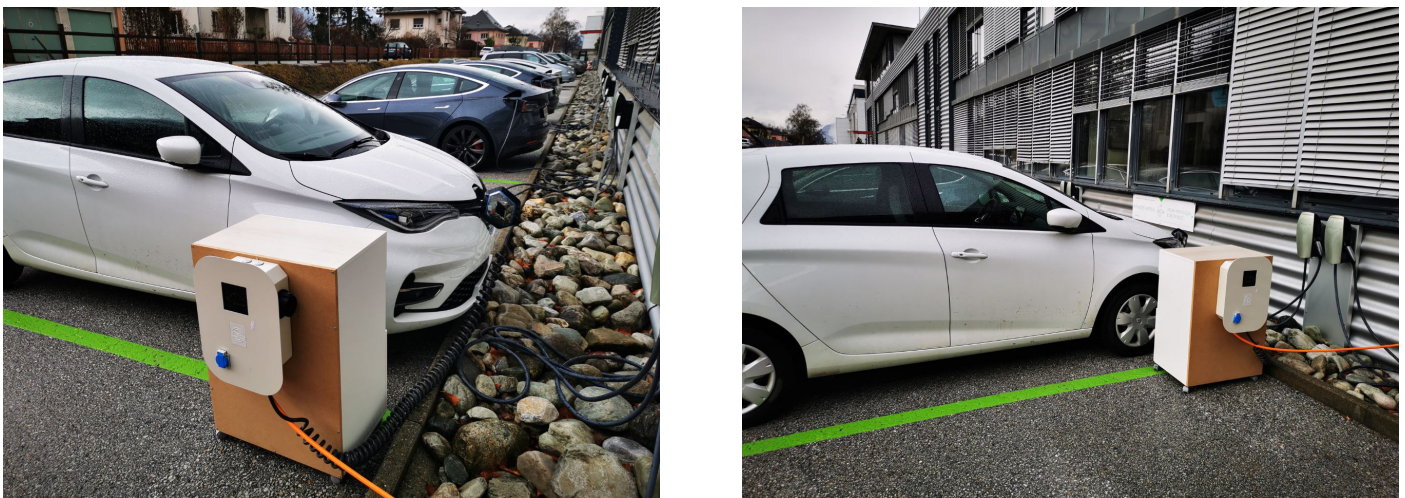


Figure 14. Real-world test infrastructure.

To enable the vehicle's participation in the blockchain-supported charging, the hotel must register the EV in the inventory using the web interface. Thus, the Hotel du Pigne that already owns a Tesla Model X, uses the 'clients tab' in their web interface's account to add a new EV. The respective dialog window shown in Figure 15 prompts the hotel manager to input the necessary information for describing the new EV.

The hotel manager makes the following entries to add the Hyundai Kona [46] to the vehicles' inventory and confirms their choices with the blue "create" button:

- **Username:** Kona
- **Password:** 1234
- **Brand:** Hyundai
- **Model:** Kona
- **Battery size [kWh]:** 64
- **AC board charger [kW]:** 11
- **DC board charger [kW]:** 77
- **Cable type:** 2 (2 = Combined Charging System (CCS), type 2)

A corresponding confirmation message depicted in Figure 16 pops up on the web interface screen to validate the new addition to the EVs' inventory. The updated list of EVs owned by the Hotel du Pigne is shown in Figure 17.

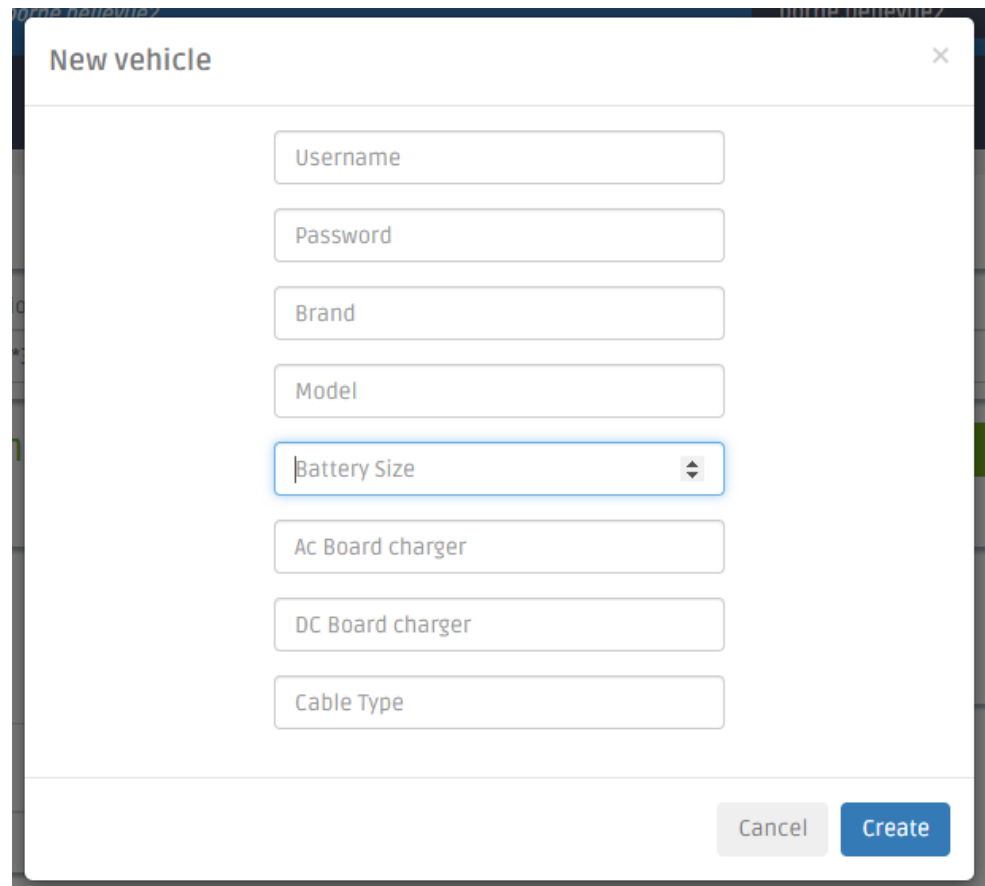


Figure 15. The “Add new vehicle” dialog window.

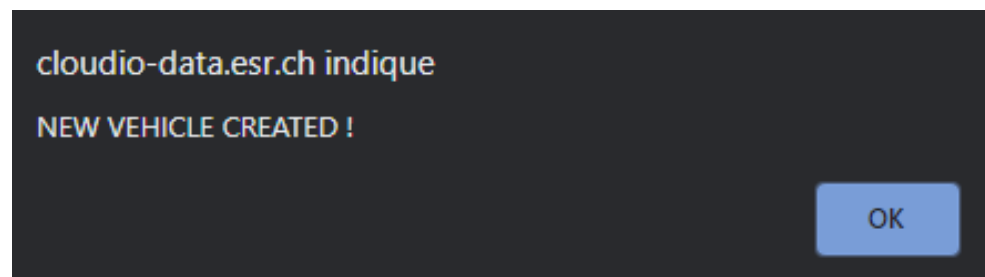


Figure 16. A confirmation message provided upon the registration of a new EV.

Once the new EV is registered, a QR code with its username and password is generated and can be shared with the hotel’s guests willing to rent the vehicle. Knowing these details allows the guests to access the mobile application and utilize it for charging management. To conduct the charging process, the user logs in to the mobile application using the provided credentials and choose the charging station of the Hotel Aiguille de la Tza as our destination. At the arrival, the EV is charged to 50%, which we optionally report in the respective field of the screen in Figure 13. Omitting the time-to-departure input, the user initiates the EV charging process by clicking on the green “start charging” button.

To form the ChargingTransaction object, at the beginning of the charging process, the charging station retrieves the following information:

- `_consumerAddress 0x150def7979a963fd24ed9b626b612f72343cedab`
- `_supplierAddress 0x3d9c273236233600b98abb4332d12f0a080b3d69`
- `_startDate 1612341928158`
- `_idVehicle 3`
- `_idChargingStation 1`

Once the charging is complete, the additional details about the charging process are included:

- `_endDate` 1612342150304
- `_energyConsumption` 0.223 [kWh]

After all the necessary information is collected, the charging station calls the `addChargingTransaction` function implemented in the smart contract. Once such blockchain transaction is validated, its record can be observed on the Etherscan [47], as shown in Figure 18.



| | | |
|---|--------------|---------|
|  | Username | Tesla |
| | Brand | Tesla |
| | Model | X |
| | Battery Size | 100 |
|  | Username | Kona |
| | Brand | Hyundai |
| | Model | Kona |
| | Battery Size | 64 |

Figure 17. The EV inventory of Hotel du Pigne.

Transactions

Latest 25 from a total of 46 transactions



| Txn Hash | Block | Age | From | To | Value | Txn Fee |
|--|---------|-------------|-------------------------|---|---------|----------------|
|  0xd72fac6c9d7d39e62... | 9590517 | 17 mins ago | 0x150def7979a963fd24... | OUT  0x69b320f9284183c0e... | 0 Ether | 0.000666092558 |

Figure 18. The record of transaction on Etherscan.

One has to note that the resulting transaction constitutes the call to the smart contract and is conducted between the hotel possessing the charging station and the smart contract itself. Thus, the hotel owning the EV does not appear as the beneficiary of the transaction due to its public address being included as the input to the `addChargingTransaction` function. Since the transaction is sent automatically by the charging station, the private key of the consumer's Ethereum account is stored locally on the charging station's computer. To ensure the security and protect the system from malicious attacks, the private key can

be encrypted to avoid being revealed to unauthorized users. The address of the deployed smart contract is referenced as follows:

- `_contractAddress 0x69B320F9284183C0E97f21a7956e6D718a62939e`

Once the transaction is validated, the charging record and the updated energy balance appear on each hotel's web interface. Therefore, as shown in Figure 19, the Hotel Aiguille de la Tza sees the last charging transaction with Hyundai Kona in green, as it was the hotel providing the energy. The same transaction appears in red in Figure 20 for Hotel du Pigne, as it was the hotel that owned the EV that consumed the energy. Although the sign varies, the total resulting energy balance of 0.15 kWh is the same for both hotels, as they were the only ones included in the testing procedure. Thus, such an energy balance signifies the correctness of the accounting and the execution of blockchain transactions.

| | | | | |
|---------------------|----------|-------|----------------|-------|
| 01/02/2021 14:15:05 | 00:00:32 | Tesla | Hotel Du Pigne | 0.016 |
| 01/02/2021 14:36:15 | 00:00:46 | Tesla | Hotel Du Pigne | 0.023 |
| 02/02/2021 08:56:26 | 00:00:55 | Tesla | Hotel Du Pigne | 0.027 |
| 02/02/2021 09:08:07 | 00:00:13 | Tesla | Hotel Du Pigne | 0.007 |
| 03/02/2021 09:45:28 | 00:03:42 | Kona | Hotel Du Pigne | 0.223 |

| |
|----------------|
| Balance |
| 0.15 kWh |

Figure 19. The web interface of the Hotel Aiguille de la Tza.

| | | | | |
|---------------------|----------|-------|--------------|-------|
| 01/02/2021 14:15:05 | 00:00:32 | Tesla | Hotel La Tza | 0.016 |
| 01/02/2021 14:36:15 | 00:00:46 | Tesla | Hotel La Tza | 0.023 |
| 02/02/2021 08:56:26 | 00:00:55 | Tesla | Hotel La Tza | 0.027 |
| 02/02/2021 09:08:07 | 00:00:13 | Tesla | Hotel La Tza | 0.007 |
| 03/02/2021 09:45:28 | 00:03:42 | Kona | Hotel La Tza | 0.223 |

| |
|----------------|
| Balance |
| -0.15 kWh |

Figure 20. The web interface of the Hotel du Pigne.

Discussion

To assess the performance of the proposed blockchain-supported EV-charging framework, we discuss the following metrics widely applied in the blockchain literature:

- **Transaction latency**, or average transaction time, is the time elapsed between the transaction's generation and its final appearance in the block on the blockchain. Despite being widely used, this metric varies strongly depending on the following

parameters: the number of simultaneous transactions on the network, the average gas price of every pending transaction, and the gas price the user is willing to pay. If the network is overloaded, users will have to set a higher gas price for their transaction to be processed and written on the blockchain by miners.

- **Transaction fee** is a cryptocurrency fee collected from users to process the transaction on the blockchain network. The fee differs depending on the complexity of the transaction, the gas price set by the user, and the price of Ethereum at the date of the transaction. This metric is related to the average transaction time since a higher transaction fee results in a shorter transaction time.
- The **contract deployment fee** is the price of the smart contract's initial deployment on the main Ethereum blockchain. This must be done once for every update of the smart contract's code. The fee works the same way as the regular transaction fee, and the price depends on the same parameters.
- The **number of nodes** is a good measure to understand the size of the blockchain network. However, it is more suitable for private blockchains. As the current work was conducted using the Ethereum public network, the total number of nodes, which currently equals 12473 on Ethereum [48], is not a metric of interest.
- **Transaction throughput** is the number of transactions per second that the network can process. Thus, it gives a good idea of how scalable the system could be in the future. However, such a metric is not applicable to evaluate our methodology's performance as we utilize the Ethereum public blockchain and are thus limited by the main network's capabilities without having the means to influence it.

To gather the data for analyzing the aforementioned metrics, we performed manual tests on the blockchain. At the time of this work, the price of a single transaction of our smart contract on the main network would be over USD 80. Therefore, for financial reasons, we only conducted our tests on the Ropsten test network.

First, we had to choose the appropriate gas price to be set for our transaction, thus indicating how fast we want our transaction to be mined. The online service [49] provides the statistics of recommended gas prices based on supply and demand for the computational power of the network needed to process smart contracts and other transactions, as seen in Figure 21. At the time of the experiment, the recommended gas price for a standard transaction was 155 Gwei, equal to 0.000000155 ETH.

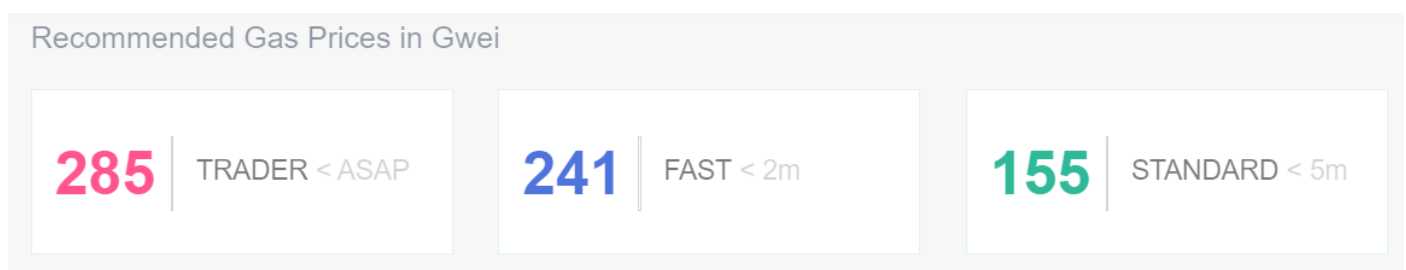


Figure 21. The recommended gas price as of 4 February 2021.

Once the gas price was defined, we had to set the gas limit to ensure that our transaction could be executed. If the computational power that is needed to execute the transaction exceeds the predefined gas limit, the transaction will be aborted as it runs out of gas. Therefore, gas limits are largely overestimated in practice to ensure the transaction's safe validation, especially as the unused gas is reimbursed. Figure 22 refers to the estimation provided by the same online service [49], which results in predicted transaction cost and mean confirmation time. In comparison to the blockchain-based energy trading platform presented in [29] and also running on public Ethereum network, our charging transaction consumes twice as much gas. However, such discrepancy in the gas utilization can be explained by additional calculations that have to be performed prior to adding

the charging transaction to the list. Moreover, the ChargingTransaction object contains a higher number of fields than the similar agreement report object in [29].

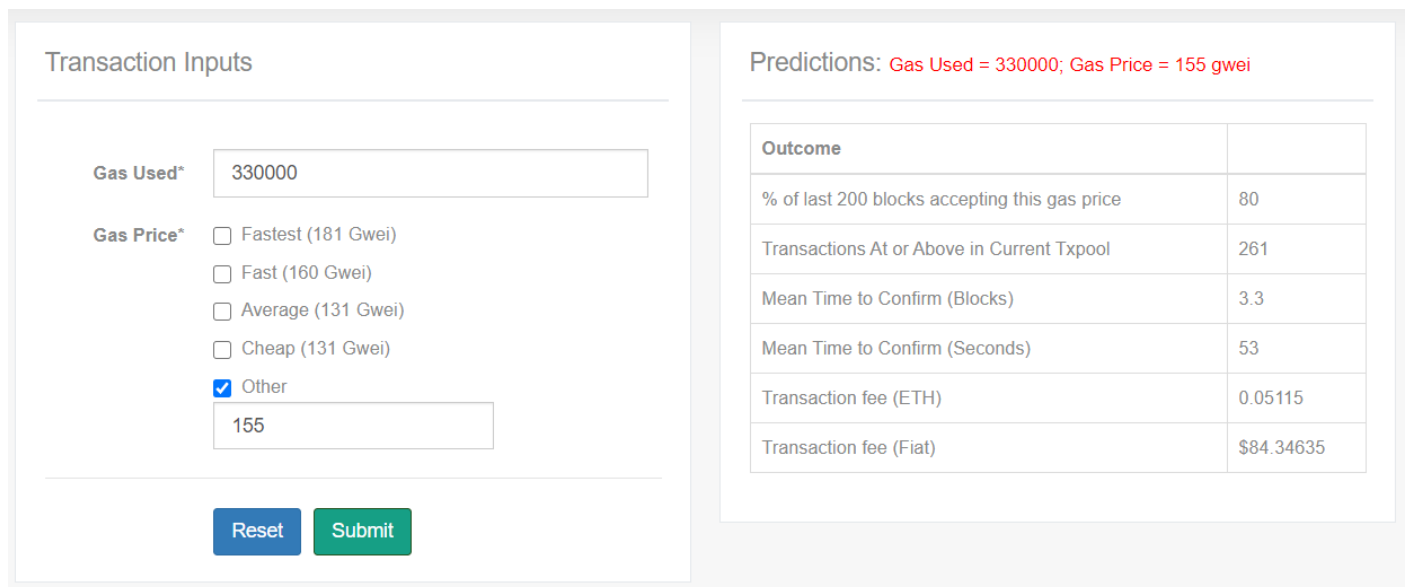


Figure 22. The gas calculator.

Table 4 summarizes the results of our experiment, particularly reporting the transaction times. The start and end of the transaction are specified as a Unix timestamp. Considering we have applied a constant gas price, the difference in transaction times can be explained by the number of simultaneous transactions in the network. Due to each Ethereum block having a maximum size of 1.5 million gas, the number of standard transactions that can fit inside is around 70. Therefore, if our transaction is processed close to the end of the block's formation, the transaction time can be fast. Otherwise, the transaction has to wait for the block's completion to be recorded on the blockchain.

Table 4. The data collected during the experiment.

| Num | Start | End | Time [ms] |
|-----|---------------|---------------|-----------|
| 1 | 1612434344629 | 1612434367000 | 22,371 |
| 2 | 1612434037000 | 1612434042535 | 5535 |
| 3 | 1612433871000 | 1612433878263 | 7263 |
| 4 | 1612433566000 | 1612433568287 | 2287 |
| 5 | 1612433265875 | 1612433297000 | 31,125 |

The metrics derived from our experiment are the following:

- **Transaction latency:** 13.716 [s]
- **Transaction fee:** 0.00074103 [ETH] \approx 1 [CHF]

The authors in [15] achieved a similar transaction latency of 16 (s); however, a different blockchain network was used during the experiment. Therefore, it might be subjected to different market dynamics governing the supply and demand for computing power. Nevertheless, all transactions on the Ropsten test network are worthless and do not reflect the price we would have paid on the main Ethereum network. Thus, the following discussion of the proposed framework's viability does not directly consider the calculated transaction fee. Moreover, the debate about the chosen blockchain's energy intensity is deliberately left out of focus as the technology, being continuously under development, is improving. As of August 2021, Ethereum successfully went through the "London Upgrade", heading to the switch from PoW to PoS consensus mechanism scheduled

for late 2021. This major change is expected to reduce costs of transactions and energy consumption, thus making blockchain applications financially and energetically viable.

The typical EV charging business model usually involves three parties, who collectively determine the price of EV charging, with their respective roles, responsibilities, and cost structures defined as follows [50]:

- **Charging station owner**, who owns the charging station and respective location. The type of ownership can be semi-public, public, and private.
- **Charging station operator (CSO)**, who is responsible for the management, maintenance, and operation of the charging station.
- **Electric-mobility service provider (EMSP)**, who offers EV charging services to end-customers.

According to [51], the cost structures of CSO and EMSP are defined in Equations (1) and (2), respectively, where C_i is the infrastructure cost related to annual amortization of charging station investment and operation and maintenance, C_e is the electricity bill, C_c is the communication cost, C_{mp} is the access to marketplace cost, C_o is the staff and overhead cost, and C_{cm} is the customer management cost:

$$C_{CSO} = C_i + C_e + C_c + C_{mp} + C_o \quad (1)$$

$$C_{EMSP} = C_c + C_{mp} + C_o + C_{cm} \quad (2)$$

One has to note that Equation (2) reflects only the EMSP's fixed costs, while the variable costs are assumed to be passed through directly to EV users [51]. Moreover, the terms in Equations (1) and (2) are used to indicate the nature of compounded costs without implying any similarity in orders of magnitude. To generate revenues and effectively compensate for their expenditures, the CSO determines the minimum average charging price, which can be both energy-based (CHF/kWh) and time-based (CHF/min), and EMSP defines the minimum value of a subscription fee, which allows the EV driver to access the EMSP's charging network [51]. Importantly, if the EV charging process is conducted outside the EMSP's network, roaming fees must be paid. At the end of the charging event, the EV driver pays EMSP, who shares its revenue with CSO, who in turn compensates the charging station owner.

As one can notice, utilization of the conventional EV charging business model leads to the accumulation of various non-charging related costs, such as C_{mp} , C_o , C_c , C_{cm} , and roaming fees. Implementing the proposed blockchain-supported EV charging management framework can potentially reduce these costs by automating the charging records procedure, impacting the role of EMSPs and influencing the rules to access the EV charging market.

To test this hypothesis, we consider the example of EVPass, one of the largest public EMSPs in the highly fragmented Swiss market [52]. The driving-related statistics required to assess the conventional business model are summarized in Table 5 along with technical details of three popular EV models: Citroen C-Zero (CCZ), Hyundai Kona (HK), and Tesla Model X Long Range (TMX). In this study, we assume that one charging event performs complete battery charging from 20% to 100%, although in practice, EV drivers prefer to top up their batteries before the minimum state of charge is reached. The number of annual recharges N_i and annual energy consumption E_i are calculated according to Equations (3) and (4), where i is the EV model:

$$N_i = \frac{D_{av} * 365}{(100\% - SOC_{min}) * D_i} \quad (3)$$

$$E_i = N_i * (100\% - SOC_{min}) * B_i \quad (4)$$

Table 5. Case study data.

| Variable | Notation | Value | Unit |
|--|-------------|-------|------|
| Minimum EV state of charge | SOC_{min} | 20 | % |
| HK battery size [46] | B_{HK} | 64 | kWh |
| HK real driving range [46] | D_{HK} | 395 | km |
| CCZ battery size [53] | B_{CCZ} | 16 | kWh |
| CCZ real driving range [53] | D_{CCZ} | 85 | km |
| TMX battery size [53] | B_{TMX} | 100 | kWh |
| TMX real driving range [53] | D_{TMX} | 440 | km |
| Average daily distance driven in Switzerland by car [54] | D_{av} | 23.8 | km |
| Number of annual recharges of HK | N_{HK} | 28 | - |
| Annual energy consumption of HK | E_{HK} | 1434 | kWh |
| Number of annual recharges of CCZ | N_{CCZ} | 128 | - |
| Annual energy consumption of CCZ | E_{CCZ} | 1639 | kWh |
| Number of annual recharges of TMX | N_{TMX} | 25 | - |
| Annual energy consumption of TMX | E_{TMX} | 2000 | kWh |

The EVPass EMSP offers three different subscription services along with a pay-as-you-go (PAYG) option. The Night, Day, and Anytime subscription packages amount to 550 CHF, 990 CHF, and 1320 CHF annually, while the first PAYG option charges a 1.5 CHF flat-rate for each charging event along with 0.5 CHF/kWh, and the second PAYG option charges 59 CHF annually along with 0.45 CHF/KWh [55]. Single-charging-event-related cost for EV driver per EV model under different payment options is calculated using the number of annual recharges and annual energy consumption determined in Table 5 and is presented in Table 6. According to the research conducted in [56], the division of revenues in the EV charging market can be approximated as follows: charging station owner 17.6%, CSO 21.2%, EMSP 7.3%, and electricity provider 53.9%. Therefore, assuming that the blockchain-supported charging system can potentially impact the role of EMSPs and reduce some of the expenses mentioned earlier, 7.3% of the charging-event costs calculated in Table 6 could serve as the reference to set cost-efficiency objectives for blockchain in EV charging.

Table 6. Charging-event related costs for EV driver in [CHF].

| EV Model | PAYG 1 | | PAYG 2 | | Night | | Day | | Anytime | |
|----------|--------|------|--------|------|-------|------|-------|------|---------|------|
| | Total | EMSP | Total | EMSP | Total | EMSP | Total | EMSP | Total | EMSP |
| HK | 27.1 | 2.0 | 25.1 | 1.8 | 19.6 | 1.4 | 35.3 | 2.6 | 47.1 | 3.4 |
| CCZ | 7.9 | 0.6 | 6.2 | 0.5 | 4.3 | 0.3 | 7.7 | 0.6 | 10.3 | 0.7 |
| TMX | 41.5 | 3.0 | 38.4 | 2.8 | 22.0 | 1.6 | 39.6 | 2.9 | 52.8 | 3.8 |

As shown in Table 6, the potential impact of implementing blockchain in the EV charging process on cost reduction varies depending on the EV model and becomes more pronounced the lower the amount of annual recharges is. In particular, in the case of CCZ, the blockchain-incurred costs would have to demonstrate higher cost-competitiveness than for HK and TMX to effectively offset the EMSP's expenditures.

Besides financial and security advantages, implementing the proposed blockchain-supported EV charging framework would result in an additional set of benefits facilitating EVs' adoption. First, simplified access to the market would give private and semi-private charging station owners the possibility to offer their charging assets to a wider public, thus increasing the size and efficiency of the EV charging network. Additionally, blockchain records' immutability would serve as a guarantee for private and semi-private owners in a trustless and often insecure environment. Second, blockchain implementation would improve the EV driver's experience by eliminating the need to manage several charg-

ing subscriptions, choosing only permitted charging stations, and paying roaming fees when charging outside the chosen network. However, one must keep in mind that using blockchain in EV charging processes is currently not fully compatible with the European Union's General Data Protection Regulation (GDPR) enforced in 2018. First, distributed data storage and processing complicate the attribution of responsibility to a single person as requested by GDPR. Second, blockchain's fundamental immutability principle contradicts the GDPR's obligation to make data editable and erasable. Therefore, despite our beliefs in data encryption solving the GDPR compliance problem, there is still a long way to go to achieve full legal compatibility.

To conclude, we note that the suggested application of blockchain technology can be adapted and extended to other use cases beyond EV charging management. For instance, the Blockchain Grid project [57] leverages blockchain in the smart grid setting to enable electricity sharing among peers by means of a decentralized platform. The authors of [58] propose a blockchain-based incentive mechanism to increase collective PV self-consumption and reduce peak demand, while the startup Sunchain [59] already provides such capabilities to their communities through an automated sharing platform compliant with the current regulatory framework [60].

5. Conclusions

In this work, we have proposed a blockchain-supported EV charging management framework that aids the participating parties in monitoring and controlling the charging process while ensuring the correct accounting of the energy flows in a trustless ecosystem. We have compared several popular blockchain implementations, such as AragonOS, Hyperledger Fabric, Energy Web Chain, and Ethereum, across comprehensive criteria and have chosen Ethereum as the framework's basis. Moreover, we have designed an EV charging-specific smart contract that governs respective energy exchanges and records the charging transactions on the blockchain. To simplify the interaction of participating entities with the suggested framework, we have designed and developed a web interface and a mobile application. Finally, we have demonstrated our framework's real-world application on the demonstration case study in Switzerland, where the charging procedure was performed between an EV and a charging station, both belonging to the hotels in the area. The conducted test has shown the viability of the proposed solution while assessing the framework's performance according to common blockchain metrics. The discussion focused on the current EV charging business model, highlighting potential cost-reduction benefits related to blockchain implementation alongside other advantages facilitating the adoption of EVs, such as access-to-market simplification for private and semi-private charging station owners. Overall, this research has helped to define the opportunities and barriers that blockchain offers in a trustless environment where energy sharing and the amount of EVs are increasing and can serve as the guideline for future blockchain implementations in the context of EV charging and smart grids.

Future work to improve the EV charging management framework and enhance the application of blockchain solutions in energy should focus on addressing the blockchain's limitations, such as high transaction costs when the network is busy, immutable transactions in case of an error, and high energy consumption implied by the consensus mechanism [61].

Specifically, further research should be conducted in four main directions:

- The system's security should be tested by conducting simulated experiences of potential cyber threats. Specifically, the reliability of storing the consumer's private key on the charging station's computer should be assessed, and various encryption methods should be tested.
- The suggested framework's scalability limits should be analyzed through a set of extensive experiments involving multiple EVs and charging stations, where the charging processes are handled simultaneously. Thus, the respective blockchain performance-related metrics should be reassessed for several scaled scenarios.

- The energy consumption information included in the blockchain transaction should be enhanced by indicating the energy content of the EV charging conducted. Particularly, the indication of respective energy shares supplied by PV and grid should be given to trace the effective usage of renewable energy.
- The current and future legal status of blockchain should be investigated in more detail to assess the policy implications beyond the already existing data protection and privacy regulations (GDPR) in the EU.

Author Contributions: Methodology, M.D., J.V., J.-M.A.; Software, M.D., J.V., J.-M.A.; Investigation, M.D., J.V., J.-M.A.; Writing—Original Draft, M.D.; Writing—Review and Editing, J.V., J.-M.A., C.B., N.W., D.W.; Visualization, M.D.; Supervision, C.B., N.W., D.W.; Funding Acquisition, N.W., D.W.; Project Administration, N.W., D.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research is part of the activities of the Swiss Centre for Competence in Energy Research on the Future Swiss Electrical Infrastructure (SCCER-FURIES), which is financially supported by the Swiss Innovation Agency (Innosuisse—SCCER program).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: The authors would like to thank Georges Darbellay (Head of Strategy and Innovation at OIKEN), without whom this work would not have been possible. The infrastructure and the collection of data realized by OIKEN made this proof of concept possible.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

| | |
|-------|---------------------------------------|
| API | Application Programming Interface |
| CCZ | Citroen C-Zero |
| CSO | Charging Station Operator |
| DAO | Decentralized Autonomous Organization |
| DApps | Decentralized Applications |
| EMSP | Electric-Mobility Service Provider |
| ESS | Energy Storage System |
| EV | Electric Vehicle |
| EWC | Energy Web Chain |
| GDPR | General Data Protection Regulation |
| HK | Hyundai Kona |
| TPS | Transactions per Second |
| TSDB | Time Series Database |
| PAYG | Pay As You Go |
| PV | Photovoltaic |
| PBFT | Practical Byzantine Fault Tolerance |
| PoA | Proof of Authority |
| PoB | Proof of Burn |
| PoW | Proof of Work |
| PoS | Proof of Stake |
| TMX | Tesla Model X Long Range |
| V2G | Vehicle-to-grid |

References

1. Di Silvestre, M.L.; Favuzza, S.; Sanseverino, E.R.; Zizzo, G. How Decarbonization, Digitalization and Decentralization are changing key power infrastructures. *Renew. Sustain. Energy Rev.* **2018**, *93*, 483–498. [CrossRef]
2. European Environmental Agency. Greenhouse Gas Emissions from Transport in Europe. 2019. Available online: <https://www.eea.europa.eu/data-and-maps/indicators/transport-emissions-of-greenhouse-gases/transport-emissions-of-greenhouse-gases-12> (accessed on 13 October 2020).
3. Henze, V. Battery Pack Prices Cited Below \$100/kWh for the First Time in 2020, While Market Average Sits at \$137/kWh. Bloomberg New Energy Finance. 2020. Available online: <https://about.bnef.com/blog/battery-pack-prices-cited-below-100-kwh-for-the-first-time-in-2020-while-market-average-sits-at-137-kwh> (accessed on 2 March 2021).
4. International Energy Agency. Global EV Outlook. 2020. Available online: <https://www.iea.org/reports/global-ev-outlook-2020> (accessed on 13 October 2020).
5. Hassija, V.; Chamola, V.; Garg, S.; Krishna, D.N.G.; Kaddoum, G.; Jayakody, D.N.K. A blockchain-based framework for lightweight data sharing and energy trading in V2G network. *IEEE Trans. Veh. Technol.* **2020**, *69*, 5799–5812. [CrossRef]
6. Hassan, N.U.; Yuen, C.; Niyato, D. Blockchain technologies for smart energy systems: Fundamentals, challenges, and solutions. *IEEE Ind. Electron. Mag.* **2019**, *13*, 106–118. [CrossRef]
7. Miglani, A.; Kumar, N.; Chamola, V.; Zeadally, S. Blockchain for Internet of Energy management: Review, solutions, and challenges. *Comput. Commun.* **2020**, *151*, 395–418. [CrossRef]
8. Andoni, M.; Robu, V.; Flynn, D.; Abram, S.; Geach, D.; Jenkins, D.; McCallum, P.; Peacock, A. Blockchain technology in the energy sector: A systematic review of challenges and opportunities. *Renew. Sustain. Energy Rev.* **2019**, *100*, 143–174. [CrossRef]
9. Bao, J.; He, D.; Luo, M.; Choo, K.K.R. A Survey of Blockchain Applications in the Energy Sector. *IEEE Syst. J.* **2020**, 1–12. [CrossRef]
10. Nakamoto, S. *Bitcoin: A Peer-to-Peer Electronic Cash System*; Technical Report; Manubot: 2019.
11. Buterin, V. A next-generation smart contract and decentralized application platform. *White Pap.* **2014**, *3*, 1–36
12. Szabo, N. Formalizing and securing relationships on public networks. *First Monday* **1997**, *2*. [CrossRef]
13. Leising, M. The Ether Thief. Bloomberg. 2017. Available online: <https://www.bloomberg.com/features/2017-the-ether-thief> (accessed on 20 October 2020).
14. Pustišek, M.; Kos, A.; Sedlar, U. Blockchain based autonomous selection of electric vehicle charging station. In Proceedings of the 2016 International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI), Beijing, China, 20–21 October 2016; pp. 217–222. [CrossRef]
15. Huang, X.; Xu, C.; Wang, P.; Liu, H. LNSC: A security model for electric vehicle and charging pile management based on blockchain ecosystem. *IEEE Access* **2018**, *6*, 13565–13574. [CrossRef]
16. Knirsch, F.; Unterweger, A.; Engel, D. Privacy-preserving blockchain-based electric vehicle charging with dynamic tariff decisions. *Comput. Sci.-Res. Dev.* **2018**, *33*, 71–79. [CrossRef]
17. Zhang, T.; Pota, H.; Chu, C.C.; Gadh, R. Real-time renewable energy incentive system for electric vehicles using prioritization and cryptocurrency. *Appl. Energy* **2018**, *226*, 582–594. [CrossRef]
18. Pajic, J.; Rivera, J.; Zhang, K.; Jacobsen, H.A. Eva: Fair and auditable electric vehicle charging service using blockchain. In Proceedings of the 12th ACM International Conference on Distributed and Event-Based Systems, Hamilton, New Zealand, 25–29 June 2018; pp. 262–265. [CrossRef]
19. Liu, C.; Chai, K.K.; Zhang, X.; Lau, E.T.; Chen, Y. Adaptive blockchain-based electric vehicle participation scheme in smart grid platform. *IEEE Access* **2018**, *6*, 25657–25665. [CrossRef]
20. Baza, M.; Nabil, M.; Ismail, M.; Mahmoud, M.; Serpedin, E.; Rahman, M.A. Blockchain-based charging coordination mechanism for smart grid energy storage units. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 504–509. [CrossRef]
21. Kang, J.; Yu, R.; Huang, X.; Maharjan, S.; Zhang, Y.; Hossain, E. Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains. *IEEE Trans. Ind. Inform.* **2017**, *13*, 3154–3164. [CrossRef]
22. Su, Z.; Wang, Y.; Xu, Q.; Fei, M.; Tian, Y.C.; Zhang, N. A secure charging scheme for electric vehicles with smart communities in energy blockchain. *IEEE Internet Things J.* **2018**, *6*, 4601–4613. [CrossRef]
23. Wang, Y.; Su, Z.; Zhang, N. BSIS: Blockchain-based secure incentive scheme for energy delivery in vehicular energy network. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3620–3631. [CrossRef]
24. Huang, X.; Zhang, Y.; Li, D.; Han, L. An optimal scheduling algorithm for hybrid EV charging scenario using consortium blockchains. *Future Gener. Comput. Syst.* **2019**, *91*, 555–562. [CrossRef]
25. Li, Y.; Hu, B. An Iterative Two-Layer Optimization Charging and Discharging Trading Scheme for Electric Vehicle Using Consortium Blockchain. *IEEE Trans. Smart Grid* **2019**, *11*, 2627–2637. [CrossRef]
26. Fu, Z.; Dong, P.; Ju, Y. An intelligent electric vehicle charging system for new energy companies based on consortium blockchain. *J. Clean. Prod.* **2020**, 121219. [CrossRef]
27. Chen, X.; Zhang, T.; Ye, W.; Wang, Z.; Iu, H.H.C. Blockchain-based Electric Vehicle Incentive System for Renewable Energy Consumption. *IEEE Trans. Circuits Syst. II Express Briefs* **2020**. [CrossRef]
28. Zhou, Z.; Wang, B.; Guo, Y.; Zhang, Y. Blockchain and computational intelligence inspired incentive-compatible demand response in internet of electric vehicles. *IEEE Trans. Emerg. Top. Comput. Intell.* **2019**, *3*, 205–216. [CrossRef]

29. Lasla, N.; Al-Ammari, M.; Abdallah, M.; Younis, M. Blockchain based trading platform for electric vehicle charging in smart cities. *IEEE Open J. Intell. Transp. Syst.* **2020**, *1*, 80–92. [[CrossRef](#)]
30. Umoren, I.A.; Jaffary, S.S.; Shakir, M.Z.; Katzis, K.; Ahmadi, H. Blockchain-Based Energy Trading in Electric Vehicle Enabled Microgrids. *IEEE Consum. Electron. Mag.* **2020**. [[CrossRef](#)]
31. Di Silvestre, M.L.; Gallo, P.; Ippolito, M.G.; Sanseverino, E.R.; Zizzo, G. A technical approach to the energy blockchain in microgrids. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4792–4803. [[CrossRef](#)]
32. Dunkan, L.; Light, J.; Cuende, L.I.; Izquierdo, J.; Spagnuolo, F. Aragon Network White Paper. 2019. Available online: <https://github.com/aragon/whitepaper> (accessed on 20 October 2020).
33. Androulaki, E.; Barger, A.; Bortnikov, V.; Cachin, C.; Christidis, K.; De Caro, A.; Enyeart, D.; Ferris, C.; Laventman, G.; Manevich, Y.; et al. Hyperledger fabric: A distributed operating system for permissioned blockchains. In Proceedings of the Thirteenth EuroSys Conference, Porto, Portugal, 23–26 April 2018; pp. 1–15. [[CrossRef](#)]
34. Energy Web Foundation. The Energy Web Chain: Accelerating the Energy Transition with an Open-source, Decentralized Blockchain Platform. 2019. Available online: <https://www.energyweb.org/reports/the-energy-web-chain/> (accessed on 2 February 2021).
35. Izquierdo, J. Aragon Chain: A Proof of Stake Blockchain for the Aragon Community. 2019. Available online: <https://blog.aragon.one/aragon-chain> (accessed on 20 October 2020).
36. Hyperledger. Hyperledger Architecture, Volume 1: Introduction to Hyperledger Business Blockchain Design Philosophy and Consensus. 2021. Available online: https://www.hyperledger.org/wp-content/uploads/2017/08/Hyperledger_Arch_WG_Paper_1_Consensus.pdf (accessed on 2 February 2021).
37. Gorenflo, C.; Lee, S.; Golab, L.; Keshav, S. Fastfabric: Scaling hyperledger fabric to 20,000 transactions per second. In Proceedings of the 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Seoul, Korea, 14–17 May 2019; pp. 455–463. [[CrossRef](#)]
38. Energy Web. EW-DOS: The Energy Web Decentralized Operating System. An Open-Source Technology Stack to Accelerate the Energy Transition. Part II, Technology Detail. 2020. Available online: <https://www.energyweb.org/reports/EWDOS-Technology-Detail> (accessed on 2 February 2021).
39. Katrenko, A.; Sotnichek, M. Blockchain Attack Vectors: Vulnerabilities of the Most Secure Technology. 2020. Available online: <https://www.apriorit.com/dev-blog/578-blockchain-attack-vectors> (accessed on 20 October 2020).
40. Chen, H.; Pendleton, M.; Njilla, L.; Xu, S. A survey on ethereum systems security: Vulnerabilities, attacks, and defenses. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 1–43. [[CrossRef](#)]
41. Zwanzger, F. Energy Web Chain—Smart Contract Cluster Analytics. 2020. Available online: <https://www.anyblockanalytics.com/blog/energy-web-chain-smart-contract-cluster-analytics/> (accessed on 2 February 2021).
42. Dorokhova, M.; Martinson, Y.; Ballif, C.; Wyrsh, N. Deep reinforcement learning control of electric vehicle charging in the presence of photovoltaic generation. *Appl. Energy* **2021**, *301*, 117504. [[CrossRef](#)]
43. Val d’Herens. Service Green Mobility. 2019. Available online: <https://www.valdherens.ch/en/service-green-mobility-fp45415> (accessed on 28 December 2020).
44. Innosuisse. Digitalisation in Energy and Mobility via SCCER. 2019. Available online: https://www.innosuisse.ch/inno/en/home/promotion-initiatives/impulsprogramm_digitalisierung.html (accessed on 28 December 2020).
45. SERI. Digitalization Report. 2020. Available online: <https://www.sbf.admin.ch/sbf/en/home/eri-policy/digitalisation.html> (accessed on 28 December 2020).
46. Electric Vehicle Database. Hyundai Kona Electric 64 kWh. 2021. Available online: <https://ev-database.org/car/1204/Hyundai-Kona-Electric-64-kWh> (accessed on 2 March 2021).
47. Etherscan. Ropsten Testnet Explorer. 2021. Available online: <https://ropsten.etherscan.io> (accessed on 2 February 2021).
48. The Ethereum Network & Node Explorer. Ethereum Mainnet Statistics, 2021. Available online: <https://www.ethernodes.org> (accessed on 2 February 2021).
49. Ethereum. Ethereum Gas Station. 2021. Available online: <https://ethgasstation.info/> (accessed on 2 February 2021).
50. Verbeek, M. Why Price Transparency in eMobility Matters. EVBox. 2020. Available online: <https://blog.evbox.com/price-transparency-emobility> (accessed on 2 March 2021).
51. Madina, C.; Zamora, I.; Zabala, E. Methodology for assessing electric vehicle charging infrastructure business models. *Energy Policy* **2016**, *89*, 284–293. [[CrossRef](#)]
52. Open Charge Map. Charging Networks in Switzerland. 2021. Available online: <https://openchargemap.org/site/country/switzerland/networks> (accessed on 2 March 2021).
53. Electric Vehicle Database. 2021. Available online: <https://ev-database.org/> (accessed on 2 March 2021).
54. Office Fédéral de la Statistique, Section Mobilité. Population’s Transport Behaviour. 2015. Available online: <https://www.bfs.admin.ch/bfs/en/home/statistics/mobility-transport/passenger-transport/travel-behaviour.html> (accessed on 2 March 2021).
55. EV Pass. The EVPass Subscription Fees. 2021. Available online: <https://www.evpass.ch/Subscription> (accessed on 2 March 2021).
56. Krug, A.; Knoblinger, T.; Saeftel, F. Electric Vehicle Charging in Europe. Arthur D. Little Switzerland. 2021. Available online: <https://www.adlittle.ch/en/insights/viewpoints/electric-vehicle-charging-europe> (accessed on 2 March 2021).
57. Energienetze Steiermark GmbH. Blockchain Grid. 2021. Available online: <https://greenenergylab.at/en/projects/blockchain-grid> (accessed on 3 October 2021).

58. Menniti, D.; Sorrentino, N.; Pinnarelli, A.; Mendicino, S.; Vizza, P.; Polizzi, G. A blockchain based incentive mechanism for increasing collective self-consumption in a nonsumer community. In Proceedings of the 2020 17th International Conference on the European Energy Market (EEM), Stockholm, Sweden, 16–18 September 2020; pp. 1–6.
59. Sunchain. 2021. Available online: <https://www.sunchain.fr> (accessed on 3 October 2021).
60. Stephant, M.; Hassam-Ouari, K.; Abbas, D.; Labrunie, A.; Robyns, B. A survey on energy management and blockchain for collective self-consumption. In Proceedings of the 2018 7th International Conference on Systems and Control (ICSC), Valencia, Spain, 24–26 October 2018; pp. 237–243.
61. Golosova, J.; Romanovs, A. The advantages and disadvantages of the blockchain technology. In Proceedings of the 2018 IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), Vilnius, Lithuania, 8–10 November 2018; pp. 1–6.