Besu vs. Quorum: Comparative Analysis in the Context of Simulated Energy Communities

Giuseppe Antonio Pierro[1], Luisanna Cocco[1] and Roberto Tonelli[1]

[1]Department of Mathematics and Computer Science, University of Cagliari, Cagliari, Italy

Abstract

This paper presents a comparative analysis of two private blockchains, Besu and Quorum, both built on the Ethereum Virtual Machine (EVM) in order to manage the simulated production, consumption and exchange of energy in the context of a Renewable Energy Community (REC).

The research focuses on simulating energy production through photovoltaic panels within a community, utilizing energy storage systems. The simulated data serve as input for two private blockchains, leveraging smart contracts to transform the simulated energy production into tokens. The study further explores the simulation of tokenized energy transactions, encompassing buying and selling within the energy community. The final phase involves a comprehensive comparison of Besu and Quorum, evaluating their computational resource usage and performance metrics.

The findings contribute to the comprehension of blockchain technologies within energy communities, offering valuable insights into the efficiency and suitability of BESU and QUORUM for tokenized energy transactions. Our research confirms that simulating an energy community scenario with 20 producers, trading energy via tokens, demonstrates no performance gap in terms of TPS and RPS between the two blockchains. However, at larger scales, Quorum appears to outperform Besu in terms of both TPS and RPS efficiency.

Keywords

BesuBlockchain, Ethereum, Proof-of-Stake, Besu, Quorum

1. Introduction

Up until about ten years ago, electricity generation was centralized, with a few large power plants fueled by fossil fuels like gas, coal, and oil spread across the country. These plants produced significant amounts of energy that was distributed through a network to users, ensuring a reliable electricity supply. This method had a high environmental cost due to the CO2 and other pollutants emitted during fossil fuel electricity production.

As a result it led in the last decade to a shift towards a decentralized model, to address climate change by promoting renewable energy sources such as wind and solar power. This new decentralized model presents challenges for the electricity grid, including managing electricity from various sources and dealing with the intermittent nature of renewable energy generation. Within this model advanced services for individual users and the coordination of various stakeholders with different roles and objectives, such as market operators and maintenance workers, are made possible through information systems and control and management systems that play a crucial role. A key element in this model is the "smart metering" system that enables remote reading and management of electricity meters, allowing for automatic and frequent monitoring from a distance. The energy system becomes more dynamic and remotely controllable, allowing network managers to intelligently oversee available electricity by monitoring real-time consumption patterns of users. This enables users to optimize their electricity usage by operating electrical appliances during periods of lower energy prices. Additionally, retailers can offer customers electricity divided into hourly bands, reflecting the varying value of electricity at

✉ antonio.pierro@gmail.com (G. A. Pierro); luisannacocco@gmail.com (L. Cocco); roberto.tonelli@unica.it (R. Tonelli)

🌐 https://web.unica.it/unica/page/it/giuseppea_pierro (G. A. Pierro); https://web.unica.it/unica/page/it/luisanna_cocco (L. Cocco); https://web.unica.it/unica/page/en/roberto_tonelli (R. Tonelli)

🆔 https://orcid.org/0000-0002-3805-7964 (G. A. Pierro); https://orcid.org/0000-0002-5055-9166 (L. Cocco); https://orcid.org/0000-0002-9090-7698 (R. Tonelli)

different times of the day and year.

Renewable energies are growing rapidly globally as found by the International Energy Agency (IEA) in its Renewables 2023 report (https://www.iea.org/reports/renewables-2023). IEA predicts that by 2028 the share of renewable electricity in Europe will be 61%, thanks in part largely to unprecedented installations of solar panels, up from the current 41%.

In this context emerging technological trends are playing a pivotal role in shaping these systems. For example Internet of things (IoT) allows for more intelligent and conscious use of energy, while Blockchain technology allows new automated payment systems, or exchanges of goods and services within the so called sharing economy.

In particular, in a context where energy production can be intermittent and also unpredictable, such as in the case of solar power, wind power, and hydropower, that rely on natural elements such as sunlight, wind, and water flow, which can vary in availability and intensity over time, the blockchain technologies combined with IoT devices can enable consumers to trade and purchase energy directly from the grid rather than from retailers leveraging also the blockchain properties of immutability for certifying production and exchange of energy in real time in a reliable framework. In addition, thanks to this technology a customer can choose a renewable energy supplier, and verify their power source thanks to the blockchain which allows to trace, moment by moment, the origin of the electricity that is supplied, from the moment of generation to the final destination of consumption.

This article was born in this context, and is part of two research projects within the framework of the Italian government's National Recovery and Resilience Plan (PNRR). Precisely, it is part of the Network 4 Energy Sustainable Transition (NEST) project, and the Ecosystem of Innovation for Next Generation Sardinia (eINS) project.

In the former project, a part of the research activities deals with the design and implementation of a digital, distributed, peer-to-peer platform enabling a local token economy among consumers/prosumers within a REC for the exchange of renewable energy among peers. In the latter project, a part of the activities deals with the design and the implementation of a more complete platform that in addition to the implementation of the local token economy just quoted, enables flexibility services and implements a local token economy in which virtuous behaviors by REC users are converted in tokens used to buy goods and services available within the REC, perfectly in accordance with the principles of a sustainable economy.

In both projects, within these research activities based on blockchain technology, the design and implementation of an adequate blockchain infrastructure play a crucial role. Both projects leverage the blockchain technology for the notarization of the energy consumption and production data in such a way that becomes impossible to alter this data without leaving a trace. In addition they leverages this technology for the conversion of the production of renewable energy and of the virtuous behaviors by REC users into tokens.

There are several different blockchain technologies available today, and all have different features and functionalities. Blockchain technologies can be broadly divided into permissioned and permissionless systems. While permissionless blockchains let anybody join and contribute without approval, permissioned blockchains require users to seek authorization before they may access and use the network. The consensus process that a blockchain uses to decide how transactions are approved and added to the ledger is another important factor. As the purpose of research activities is to build a blockchain infrastructure and, consequently, the deployment of a trustworthy and efficient blockchain solution, a few crucial indicators must be assessed in order to ascertain how well these blockchain platforms operate. These consist of transaction throughput, hence the speed at which legitimate transactions are added to the blockchain, transaction latency, hence the length of time a transaction must remain in the network after it is sent, and scalability. Thus the optimal Blockchain network infrastructure for the specific use cases in these projects must be selected by carefully analyzing these indications, which provide crucial insights into the capabilities and limitations of these different blockchain technologies. This article focuses on this key part of the activities.

Precisely aim of this article is the comparison between the performance of a Blockchain infrastructure realized in Hyperledger Besu technology and a Blockchain infrastructure realized in Quorum technology.

The two networks implement two different mechanism of consensus. The former uses the Istanbul Byzantine Fault Tolerance (IBFT) algorithm and the latter uses the Replicated State Machine Approach (RAFT) algorithm. A concrete evaluation in a simulated scenario of the different blockchain technologies can help in determining which technology is best suited in real world use cases of energy production and exchange in a Renewable Energy Community where blockchain performances are a key factor to ensure reliability of the entire framework.

2. Background

In the realm of blockchain technology, the performance evaluation of the different blockchain technologies holds significant importance for organizations seeking to implement robust and efficient blockchain solutions. The evaluation of the performance of these platforms involves assessing various key metrics such as transaction throughput that is the rate at which valid transactions are committed to the blockchain, transaction latency that is the amount of time a transaction requires to be part of the network from the moment it was submitted, and scalability. By conducting a thorough analysis of these metrics, organizations can gain valuable insights into the capabilities and limitations of these technology, enabling them to make informed decisions regarding the selection of the most suitable platform for their specific use case. Among the several blockchain technologies, Hyperledger Besu and Quorum are both widely recognized for their capabilities in facilitating secure and scalable blockchain networks and this article focuses on evaluating the performance of these two blockchain under two different consensus algorithms, specifically Istanbul Byzantine Fault Tolerance 2.0 (IBFT 2.0) for Hyperledger Besu and Replicated State Machine Approach (RAFT) for Quorum have been selected for our use case. In the following a brief description of these two blockchains and of the consensus algorithms selected is reported.

2.1. Hyperledger Besu and IBFT 2.0

Hyperledger Besu is an open source Ethereum client developed by ConsenSys. It is designed to be interoperable with other blockchain technologies and to support the so-called smart contracts, i.e. contracts that are automatically executed when terms and conditions written directly in the code are met. Hyperledger Besu supports the Solidity programming language, which is the most widely used language for developing smart contracts on Ethereum, and also supports Vyper a language designed to improve the security and auditability of smart contracts compared to Solidity. It supports the Ethereum Virtual Machine (EVM) and the Web3 API, and therefore it supports the interaction with smart contracts deployed on other Ethereum-based networks. Among its main features there is the support for private transactions, which allow the use of the platform without exposing sensitive data and its ability to process thousands of transactions per second. Hyperledger Besu supports multiple consensus algorithms, therefore different selection possibilities for the set of validators that create new blocks and validate transactions, thus allowing the choice of the approach that best suits the needs of each use case. Among the algorithms Proof of Authority (PoA), proof of work (PoW), proof of stake (PoS), and IBFT can be quoted. Let us give a brief description of the consensus mechanism IBFT 2.0.
It is built upon the IBFT for addressing the safety and liveness limitations of IBFT [1]. As every blockchain node, each IBFT 2.0 node maintains a local copy of the blockchain, and the first block, is called genesis block. Each block B in the chain is cryptographically linked to another block in the chain. In IBFT 2.0 the block to be added to the local blockchain, that is equal in every network node, represents the child of the latest block that was added to the blockchain. So, the IBFT 2.0 blockchain is a linked list of blocks, in which each block has a height h, defined as the number of links between that block and the genesis block that has height h equal to 0. The IBFT 2.0 protocol can be modeled as running sequential instances, that have to decide which Ethereum block, has to be added at height h of the chain. Only a subset of nodes can participate in the $h\text{-}th$ instance, hence only a subset of nodes can contribute to take this decision. These nodes are called the validators of a specific block, hence related to a specific height/instance h. All the other nodes that do not participate in the $h\text{-}th$ instance are standard nodes.

The set of validators is deterministically computed as function of the chain of blocks from the genesis block until the block with height *h-1*. For each instance there is a round, in which one of the validators has to propose an Ethereum block for the height associated with the specific instance that he is running. Once agreement is reached, a finalized block is created. It includes the Ethereum block and additional information that allows all nodes in the network to verify that the finalised block was correctly added. To add or remove a validator from the set relative to a specific block, validators for that block can vote. The validator is added or removed from the set when more than half of the validators cast a consistent vote.

2.2. Quorum and RAFT

Quorum is an open source project launched in 2016. It is a variant of the Ethereum blockchain designed for an enterprise audience. GoQuorum is an open-source variant of Quorum, which is a permissioned blockchain platform developed by JPMorgan Chase. Both platforms are based on the Ethereum protocol and are designed for enterprise use cases. Both are lightweight forks of geth and are updated as geth releases occur. Like Besu, Quorum supports private transactions and thanks to organization registries, allowlists, and permissioning contracts it allows you to manage different authorization levels that can be customized for different roles. It supports the solidity language for writing smart contracts and is mainnet compatible. In Quorum the consensus protocols that can be used are based on voting algorithms which do not require computational power, for example IBFT, or RAFT.

Let us give a brief description of the consensus mechanism RAFT [2], that is the mechanism adopted in this article for the Quorum blockchain. The RAFT consensus algorithm distributes a state machine across a cluster of computing systems to ensure that all nodes in the cluster agree on the same set of state transitions, even in the event of failures. RAFT operates by electing a leader within the cluster, with servers functioning as leaders, followers, or candidates if the leader is unavailable. The leader oversees registry replication on other servers and handles client requests until a new leader is elected in case of leader failure or disconnection. RAFT operates through a series of phases, including leader election, log replication, and safety properties to maintain consistency and fault tolerance. With Leader Election RAFT ensures a new leader is elected when the current leader fails, with Log Replication, RAFT maintains synchronization between the leader's server and other servers' logs, and with the safety property prevents other servers from requesting a registry entry at a specific index if a server has already committed an entry for that index to maintain log consistency and execute the same commands.

3. Related Work

The concept of a smart grid has emerged as a modernized version of the traditional power grid, aiming to effectively incorporate green and renewable energy technologies. The primary objective of the grid is to establish a sustainable society. The smart grid is currently transitioning from a centralized structure to a decentralized topology, and blockchain technology offers promising features for its application in the smart grid concept. Numerous literature articles explore this topic in depth.

Mollah et al. [3] provided a comprehensive survey on the application of blockchain in smart grid, identifying the security challenges of smart grid scenarios that can be addressed by blockchain. Musleh et al. [4] examined benefits, methodologies, and technical obstacles associated with integrating blockchain technology into the smart grid, introducing frameworks for blockchain-based applications in the smart grid and demonstrating how blockchain can serve as the cyber-physical layer of the smart grid. Mehdinejad et al. [5] presented a design and model for a completely decentralized peer-to-peer (P2P) energy token market tailored for small-scale prosumers within a smart grid setting leveraging blockchain technology. Agung et al. [6] leveraged blockchain technology as a means to oversee transactions within the smart grid. Transactions are executed using smart contracts, with the network serving as a verifier for these transactions ensuring the permanence of transactions, and guaranteeing the execution of every transaction between generators and consumers. Doan et al. [7] presented a peer-to-peer (P2P) energy trading system involving prosumers utilizing a double auction-based game theory approach

and blockchain technology to showcase the viability of real-time P2P trading. In this system, the buyers adjust their energy purchase based on fluctuating electricity prices to optimize their benefits, the auctioneer oversees the game, while the seller does not actively participate but ultimately achieves maximum social welfare.

An interesting web article entitled *Blockchain technology for the renewable energy sector: A comprehensive study* [8] quotes the Brooklyn MicroGrid project, began in April 2016 (see web site https://brooklynmicrogrid.com/) as the first real application of blockchain in RE. This project is based on Ethereum blockchain technology and stores energy transactions. By utilizing smart contracts based on Ethereum, the system enables customers to purchase extra renewable energy from prosumers through a tokenized transaction system. Surplus energy generated by prosumers using rooftop solar panels is transformed into tokens by smart meters installed in their residences, which can then be utilized for direct trading in the energy market. This article investigated Blockchain use cases in the energy sector, showing that 33% use cases is associated with decentralized energy trading, and 19% use cases is associated with cryptocurrencies, tokens, and investments. Further, it classified Blockchain activities in the energy sector according to the platform used and consensus algorithms, reporting that 60% of applications are Ethereum-based solutions, and 55% adopt PoW algorithms. Furthermore, it claimed that the trend, that is developing, is oriented towards private permissioned platforms that are most attractive to enterprises. Also in our work, private permissioned platforms are selected as the more suitable technology to develop blockchain based applications for REC, and specifically our choice of Blockchain infrastructures fell on Hyperledger Besu and Quorum.

In recent years, there has been a growing body of literature focusing on the evaluation of blockchain technology performance. Monrat et al. [9] conducted a study of the performance and scalability of four private blockchains, precisely of Ethereum, Quorum, Corda, and Hyperledger Fabric, by varying the number of transactions and nodes and evaluating metrics such as throughput and network latency. Hao et al. [10] proposed a method to evaluate the performance of some consensus algorithms in Ethereum and Hyperledger Fabric private blockchain. The authors demonstrated that the PBFT algorithm surpasses the PoW algorithm in terms of both average throughput and average delay, with the performance gap widening as the number of transactions grows. Mostarda et al. [11] presented a performance analysis of a permissioned blockchain in a real scenario in which validator nodes are geographically distant from each other, and proposed a new benchmarking tool for Ethereum-like blockchains. Baliga et al. [12] studied the throughput and latency characteristics of Quorum with different workloads and RAFT AND IBFT consensus algorithms. Their findings revealed that implementing the RAFT algorithm results in minimal changes to the system's throughput when extending the block time, although latencies exhibit a proportional increase. In terms of throughput, RAFT and IBFT are similar, with RAFT demonstrating slightly superior performance at transaction rates exceeding 1650 tx/sec, while IBFT outperforms RAFT marginally at lower transaction rates. In addition they showed that private contracts exhibit lower throughput under heavier system loads, attributed to the additional overhead associated with secure communication, than public contracts, particularly when the input transaction rate exceeds 600 tx/sec. The system's maximum sustainable load was found to be 900 tx/sec, beyond which consensus could not be reached. Abdella et al. [13] proposed a permissioned Blockchain platform for Peer-to-Peer energy trading using the IBFT consensus algorithm. This architecture operates on an auction mechanism, requiring both consumers and producers to compete against each other to secure the auction win, and through several smart contracts implements three different types of energy markets, payment settlement and authentication functions. The authors developed this platform both using Hyperledger Besu and Hyperledger Fabric blockchin platforms, and compared the performance of the IBFT algorithm with Clique, PoW, and Raft consensus algorithms By analyzing transaction latency, transaction throughput and fail rate they showed that IBFT has better performance.

4. Research Questions and Hypotheses

The ever-growing demand for high-performance blockchain solutions necessitates a clear understanding of different platforms' capabilities. This research focuses on evaluating two prominent Ethereum-based blockchains, Quorum and Besu, in terms of their ability to handle transaction and request processing [14, 15].

This section outlines the key research questions that will guide our investigation and the hypotheses we aim to test. By addressing these questions, we hope to gain valuable insights into the performance characteristics of Quorum and Besu.

The following research questions serve as the foundation for our investigation:

4.1. Research Questions:

1. Which blockchain platform, Quorum or Besu, demonstrates a superior capability in processing transactions per second (TPS) and requests per second (RPS)? This primary question aims to identify the platform that offers faster transaction and request handling.
2. How do the scalability characteristics of Quorum and Besu compare under varying workloads and network conditions? Evaluating scalability is crucial, as real-world use cases can involve fluctuating transaction volumes and network pressures.

4.2. Research Hypothesis:

1. Null Hypothesis (H0): There is no significant difference in the TPS and RPS processing capabilities between Quorum and Besu blockchains.
2. Alternative Hypothesis (H1): There is a significant difference in the TPS and RPS processing capabilities between Quorum and Besu blockchains. We expect one platform to outperform the other, and this research aims to identify which and under what conditions.

5. Research Methodology

5.1. Set-up of the experimental system

The two blockchains were set up with identical hardware resources but different software configurations due to their distinct consensus mechanisms. Concerning the hardware resources, we utilized two Apple 2020 Mac mini with Chip M1 (8GB RAM, 256GB SSD), and a MacBook M2 with 16GB RAM and 500GB SSD. Regarding the software configuration, it is imperative to differentiate between Besu and Quorum blockchains.

In the software configuration of the Besu blockchain, we chose to deploy a network consisting of five nodes, with four nodes serving as validators and one node fulfilling the role of an RPC node. Specifically, Mac Mini devices were configured in a manner to host two virtual machines each, with each virtual machine serving as a validator. The machine equipped with the M2 chip was utilized as the RPC node. On each node, we installed Node.js version 20 and Java version 18. Java is a high-level designed to be platform-independent, meaning that Java programs can run on any device or operating system that has a Java Virtual Machine (JVM) installed.

In the software configuration of the Quorum blockchain, each node is standardized with identical software characteristics. Specifically, on machines equipped with M1 chips, we deployed two virtual machines, while on machines equipped with M2 chips, a single virtual machine has been instantiated. Moreover, across all nodes, we implemented version 20 of both Node.js and GoQuorum. This ensures uniformity and consistency in the software environment across the entire blockchain network. GoQuorum is an open-source Ethereum client developed by ConsenSys. It is based on the Go Ethereum (Geth) client and is specifically designed for use in private or consortium blockchain networks. GoQuorum provides features such as privacy, permissioning, and high performance, making it suitable for enterprise blockchain applications.

In order to simulate the production transactions of electricity generated by photovoltaic panels, we employed a Node.js script that leverages the Web3 library. This script was utilized for conducting simulations on both the Quorum and Besu blockchain platforms. For each simulated energy production event, we correspondingly incremented the token associated with the owner's address. By employing this script across both blockchain platforms, we ensured methodological consistency in our simulations and facilitated a seamless comparison of results between the two environments.

In order to conduct stress tests and ascertain the maximum transaction throughput that both blockchains could handle, we employed different tools tailored to each blockchain environment. This approach allowed us to simulate higher transaction volumes and evaluate the scalability of each blockchain platform under varying loads. Table 1 provides a comprehensive overview of the testing tools employed to conduct stress tests on the Besu and Quorum blockchains.

Table 1
Testing Tools for Blockchain

Testing Tool Name	Blockchain Name	Goal	Link
Hyperledger Caliper tool	Besu	Transactions Per Second (TPS)	https://github.com/hyperledger/caliper
Apache JMeter	Quorum	Transactions Per Second (TPS)	https://github.com/apache/jmeter
Ethspam	Besu	JSON RPC requests	https://github.com/shazow/ethspam
Chainhammer	Quorum	JSON RPC requests	https://github.com/drandreaskrueger/chainhammer

To evaluate the maximum achievable Transactions Per Second (TPS) on our Besu blockchain, we employed the open-source Hyperledger Caliper tool. This industry-standard performance benchmarking framework, available at https://github.com/hyperledger/caliper, is specifically designed for assessing the capabilities of various blockchain platforms. Hyperledger Caliper facilitates comprehensive performance analysis by generating detailed reports after each benchmark run. These reports capture critical metrics such as TPS, transaction latency, resource utilization (CPU and memory), and transaction success/failure rates. This valuable data empowers us to make informed decisions regarding our Besu network's performance optimization, pinpointing potential bottlenecks and areas for improvement.

We leveraged Apache JMeter, an open-source performance testing tool, to assess the maximum achievable Transactions Per Second (TPS) on our Quorum blockchain network. JMeter's platform-independent design, facilitated by its pure Java implementation, enabled us to conduct these tests seamlessly across various operating systems. While offering a user-friendly graphical user interface (GUI) for test plan creation, JMeter empowers advanced users with scripting capabilities using languages like BeanShell and JavaScript. This flexibility allowed for tailored automation and customization of test logic, ensuring a comprehensive evaluation of our Quorum network's transaction processing capacity.

To assess the maximum capability of Besu and Quorum blockchains in handling JSON RPC requests, we utilized two distinct tools. This phase allowed us to establish the upper bounds of processing capacity for both Besu and Quorum blockchains It is pertinent to acknowledge that RPC serves as a well-established communication protocol, facilitating applications in making remote procedure calls on a server. JSON (JavaScript Object Notation) has emerged as the preferred format for data exchange between applications engaged in RPC calls. Consequently, an RPC JSON request embodies a message formatted in JSON, transmitted to a server with the objective of invoking a specific function or procedure.

For the purpose of evaluating the Besu blockchain, we selected and implemented a tool known as Ethspam. This tool is specifically designed to generate a continuous stream of realistic Ethereum JSON RPC queries. It is noteworthy that these queries are meticulously crafted to be read-only in nature, signifying that they are not intended to induce any modifications to the state of the blockchain. To comprehensively evaluate the Quorum blockchain, we opted to leverage a tool designated as Chainhammer. By employing these aforementioned tools, we were able to conduct a comprehensive

and insightful analysis of the Besu and Quorum blockchains' ability to process JSON RPC requests. This in-depth exploration yielded invaluable insights into their performance characteristics, empowering us to make informed decisions regarding their suitability for our specific requirements.

5.2. Simulation of Energy Production from Photovoltaic Panels

This section provides a detailed explanation of how the data regarding electrical energy produced through photovoltaic panels was generated. We conducted simulations to determine the quantity of electrical energy stored in the storage batteries and produced by the photovoltaic panels across different time intervals throughout the day. The simulation builds upon a previous study that provides data on the performance estimation of photovoltaic energy production [16]. Specifically, the data from this research were generated using a specialized program designed to accept a list of objects categorized as "producers" as input values. Within this framework, a "producer" object encompasses various parameters that play a role in the simulation process, as outlined in Table 2. From a simulation perspective, each "producer" object is characterized by a multitude of properties, including but not limited to, the nominal capacity for electrical energy production of the system and the technological specifications utilized by the inverters. Table 2 serves to delineate and elaborate on all pertinent parameters that are configurable within the simulator framework for photovoltaic energy production.

Producer Parameters	Description
Battery Capacity	The maximum amount of electrical energy that can be stored in the battery.
Battery Efficiency	The ratio of usable energy output to the total energy input into the battery.
Battery Type	The specific type or technology of the battery used in the system.
Inverter Efficiency	The percentage of input electrical energy converted to usable output energy by the inverter.
Panel Efficiency	The efficiency of converting sunlight into electricity by the photovoltaic panels.
Nominal Power	The rated power output of the photovoltaic system under standard test conditions.
Panel Orientation and Tilt	The angle and direction at which the photovoltaic panels are installed relative to the sun.

Table 2
Producer Parameters and Descriptions

The figure 1 depicts the energy generated by a consumer over the course of a week. As illustrated, the curve remains flat during nighttime hours and increases, reaching its peak at noon. It is noticeable that on two particular days, the curve does not exhibit a Gaussian shape due to several hours of cloudy skies, resulting in a reduction of electrical energy production by approximately 80%.

Listing 1 showcases part of a larger program aimed at simulating energy production from photovoltaic panels over the course of a week. The purpose of this specific segment is to generate data representing the energy production at different times throughout the week. It iterates over each day, hour, and minute, applying various conditions to simulate the energy production process, including fluctuations due to weather conditions and time of day.

Listing 1: energy production data

```
1  # Generate energy production data
2  for day_idx, day in enumerate(days):
3      for hour in range(day_hours):
4          for minute in range(0, 60, sampling_interval_minutes):
5              idx = (day_idx * day_hours * 2) + (hour * 2) + (minute //
                      sampling_interval_minutes)
6              if hour in night_hours:
7                  energy_production[idx] = 0  # No solar energy production during
                          the night
```

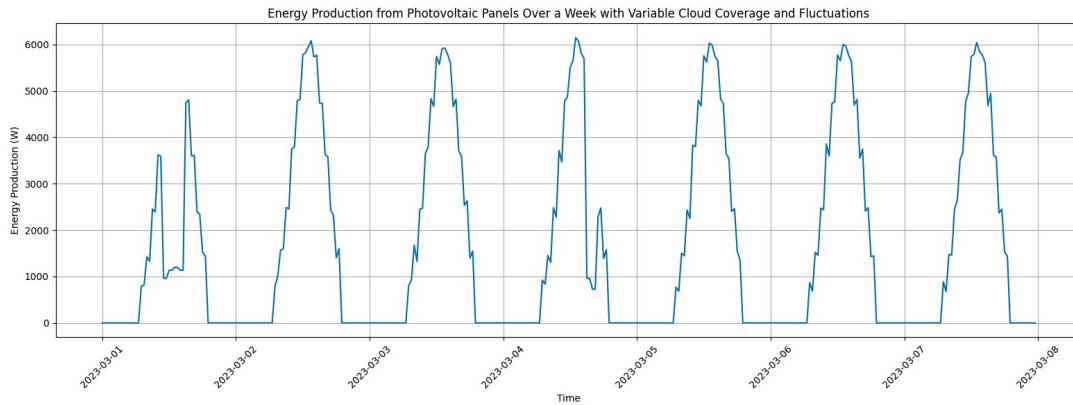**Figure 1:** Energy production over a week

```
8                    else :
9                        # Introduce slight fluctuations for both sunny and cloudy days
10                       base_production = 6000 * np.exp(-0.5 * (( hour - peak_hour ) / 3)
                             ** 2)

12                       # Additional conditions for reduced energy production on
                             specific days and hours
13                       if ( day == 'Monday' and 11 <= hour < 15) or ( day == 'Thursday'
                             and 15 <= hour < 17):
14                           energy_production[ idx ] = 0.2 * base_production
15                       else :
16                           variability = np.random.normal( loc =0 , scale =100)
17                           energy_production[ idx ] = max(0 , base_production +
                                 variability )
```

5.3. Energy Token Exchange Smart Contract

This section describes the smart contract employed specifically for the comparative analysis performed in this paper. When energy is produced, the contract triggers the creation (minting) of energy tokens, with their quantity incremented to reflect the energy generated. Conversely, when energy is consumed, the contract reduces the quantity of tokens accordingly. This meticulous tracking mechanism ensures a precise and up-to-date representation of the available energy within the network, enabling efficient management and utilization of energy resources.

Note that the smart contract, implemented in the projects to which this paper refers, implements a more sophisticated management system designed to monitor and record the ownership of energy as well as the exchange of energy among peers in the REC. Within the community each participant is associated with one or more smart meters. Each participant and smart meter is associated with a specific address, and only the owner of the energy token, or an address authorized, can manage the token for example incrementing or decrementing its stored energy, or for example selling the token thus transferring the token's ownership from a peer to another one.

Listing 2 presents the Solidity smart contract utilized in this article. The smart contract facilitates the creation and management of tradable tokens representing energy units. Built upon the ERC721 standard, it enables users to interact with energy-based assets in a secure and transparent manner.

With regard to the Token Minting and Energy Management aspects of the smart contract, it is worth highlighting the following key points:

- The mint function serves as the cornerstone of token creation. It allows authorized entities (governed by access control mechanisms) to mint new EnergyToken instances. These tokens are assigned to designated addresses and carry an initial energy amount.

- The incrementEnergyAmount function caters to scenarios where additional energy needs to be credited to a specific token. This functionality is crucial for situations like energy generation or purchase.
- Conversely, the decrementEnergyAmount function facilitates energy consumption or transfer by reducing the associated value for a particular token. It ensures a balance check to prevent exceeding the available energy units.

In terms of internal functionality and security, the smart contract utilizes two critical mappings to facilitate efficient data management.

- energyTokens: This mapping serves as the central repository for energy data. It associates each token ID with its corresponding energy amount.
- _tokenExists: This internal mapping plays a critical role in ensuring data integrity. It efficiently tracks the existence of tokens, preventing unnecessary checks and enhancing security.
- The _exists function serves as an internal helper, not directly accessible from outside the contract. It provides a secure way to verify token existence within the contract's logic.

Regarding the Benefits and Use Cases of this smart contract, they can be summarized in the following key points:

- Representing and managing tradable energy units on a blockchain platform.
- Enabling secure and transparent energy transactions.
- Facilitating efficient energy tracking and consumption monitoring.

By leveraging the immutability and transparency of blockchain technology, this contract empowers the development of innovative energy markets and fosters trust in energy-related transactions.

Listing 2: Energy Token Exchange Smart Contract

```solidity
1  // SPDX-License-Identifier: MIT
2    pragma solidity ^0.8.20;
3    import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
4
5    contract EnergyToken is ERC721 {
6        uint256 public tokenIdCounter;
7        mapping(uint256 => EnergyTokenStruct) public energyTokens;
8        mapping(uint256 => bool) private _tokenExists;
9
10       struct EnergyTokenStruct {
11           uint256 energyAmount;
12       }
13
14       constructor(string memory _name, string memory _symbol) ERC721(_name,
             _symbol) {}
15
16       modifier onlyOwnerOfToken(uint tokenId) {
17         require(
18           _ownerOf(tokenId)==msg.sender,
19           "You must be the owner of the token in order to manage it."
20         );
21         _;
22       }
23
24       function mint(uint256 _energyAmount) public {
25           _mint(msg.sender, tokenIdCounter);
26           energyTokens[tokenIdCounter] = EnergyTokenStruct(_energyAmount);
27           _tokenExists[tokenIdCounter] = true;
28           tokenIdCounter++;
29       }
30
```

```
31        function incrementEnergyAmount(uint256 _tokenId, uint256 _increment)
              onlyOwnerOfToken(_tokenId) public {
32            require(_exists(_tokenId), "Token ID does not exist");
33            energyTokens[_tokenId].energyAmount += _increment;
34        }
35
36        function decrementEnergyAmount(uint256 _tokenId, uint256 _decrement)
              onlyOwnerOfToken(_tokenId) public {
37            require(_exists(_tokenId), "Token ID does not exist");
38            require(energyTokens[_tokenId].energyAmount >= _decrement, "
                  Insufficient energyAmount");
39            energyTokens[_tokenId].energyAmount -= _decrement;
40        }
41
42        function _exists(uint256 _tokenId) internal view returns (bool) {
43            return _tokenExists[_tokenId];
44        }
45    }
```

5.4. Enabling Energy Token Transactions

This section elaborates on the script utilized for simulating token increment and decrement transactions within the energy token system. Within our simulated environment, we posited that energy storage operations within various batteries would occur at a frequency of one operation every fifteen minutes. Each update from sensor readings triggers a transaction via the deployed smart contract to update the token tracking energy.

Technical Implementation: The script leverages the Web3.js library to interact with the deployed smart contract. It simulates transactions by invoking the incrementEnergyAmount and decrementEnergyAmount functions based on incoming sensor data. This approach ensures a realistic simulation of energy token transactions in a real-world context.

Benefits and Significance: This simulation tool serves as a valuable resource for developers to test and refine the functionality of the smart contract. It aids in estimating the gas costs associated with different transaction types and provides insights into the overall performance and scalability of the energy token system.

Future Enhancements: The script can be extended to accommodate additional parameters, such as energy prices and transaction fees, thereby enhancing its utility. Furthermore, integration with other blockchain-based applications can create a comprehensive energy management system.

The transaction simulation script presented here offers a valuable means of evaluating and optimizing the functionality of the energy token system. By replicating real-world scenarios, it empowers developers to enhance the system's efficiency, security, and scalability.

The frequency of energy storage operations should be replaced with the actual value utilized in the system. The technical implementation details can be further elaborated upon to address specific project requirements. Similarly, the benefits and significance of the simulation tool can be tailored to suit the intended use case and target audience.

6. Results

In this research, we simulated two types of use cases to evaluate the performance and scalability of our proposed blockchain-based energy trading platform.

The first use case focused on a small-scale scenario with 20 prosumers (producers and consumers) equipped with photovoltaic panels. These prosumers traded their surplus energy and battery storage capacity using tokens on the blockchain. This scenario aimed to assess the feasibility and efficiency of our platform in a realistic setting with a limited number of participants.

The second use case aimed to test the limits of our technical solutions, implemented on both Besu and Quorum blockchains. To achieve this, we employed open-source tools designed to simulate the

maximum number of transactions per second (TPS) that each blockchain can handle, as well as the number of JSON-RPC requests that can be processed to retrieve information from the blockchain blocks. By simulating a large number of concurrent users and transactions, we were able to identify potential bottlenecks and performance limitations in our platform.

6.1. Simulation with 20 Prosumers

This section provides a detailed exposition of the results obtained from simulating electricity production and consumption as transactions recorded on the two respective blockchains, Besu and Quorum, within the energy community. The outcomes are meticulously captured in tabular formats and graphical representations, meticulously designed to facilitate a comprehensive interpretation and analysis of the data. Through these structured presentations, stakeholders can gain valuable insights into the dynamics of energy production and consumption within the community, enabling informed decision-making and strategic planning.

The metrics we elected to measure in order to assess the capacity of the two blockchains to meet the needs of an energy community are Transactions Per Second (TPS) and Requests Per Second (RPS). Both TPS and RPS parameters are influenced by various factors, including the number of nodes utilized as validators and the number of nodes used as RPC.

6.1.1. Transactions Per Second

Transactions Per Second (TPS), in the context of blockchain technology, refers to the measurement of the rate at which transactions are processed and confirmed on a blockchain network within a given timeframe, typically one second. It quantifies the network's capability to handle a certain volume of transactions within a specific duration. In simpler terms, TPS indicates how many transactions can be executed and recorded on the blockchain per second. It's a crucial metric for evaluating the scalability and performance of a blockchain network. Higher TPS values generally imply that the network can handle more transactions efficiently, while lower TPS values may indicate limitations in scalability and processing speed. Achieving high TPS is often a key objective for blockchain developers and network operators, especially in applications where real-time transaction processing is essential, such as financial transactions or supply chain management.

6.1.2. Requests Per Second

Requests Per Second (RPS), in the context of blockchain and JSON RPC (Remote Procedure Call) requests, refers to the number of requests sent to the blockchain network per second. In a blockchain system, JSON RPC is often used as a method for clients to communicate with the blockchain network. These requests can include various actions such as querying data, sending transactions, or interacting with smart contracts. RPS measures how many of these requests the blockchain network can handle within a one-second time frame. It indicates the network's capability to process incoming requests efficiently and in a timely manner. Higher RPS values typically indicate a more scalable and responsive blockchain network, capable of accommodating a larger volume of user interactions.

6.2. Stress Test Results under Maximum Load

The results section pertaining to the stress test is divided into two distinct parts. The initial part delves into the examination of the system's utmost capability in managing JSON RPC requests. Subsequently, the latter part scrutinizes the system's maximum capacity in handling the volume of transactions per second (TPS).

6.2.1. Evaluating Besu's Performance under Stressful JSON RPC Loads using Ethspam

To assess the ability of Besu to handle JSON RPC requests, we utilized the ethspam library, available at the following address: https://github.com/shazow/ethspam. Ethspam generates a continuous stream of

authentic read-only Ethereum JSON RPC queries, centered around the latest block with a certain degree of random variance. The most recent state is refreshed every 15 seconds, ensuring continuous operation without data becoming outdated. On average, the Ethspam application produces an approximate output of 500,000 lines or 120 megabytes of data per second on the designated computer desktop utilized for our testing purposes. In the performance evaluation, Hyperledger Besu demonstrated a robust ability to handle high transaction volumes of JSON RPC requests. When subjected to a simulated workload of 100,000 JSON RPC requests, Besu was able to process and respond to an average of 2,201.01 requests per second. This translates to processing over two thousand transactions every second, showcasing Besu's potential for high-throughput applications.

6.2.2. Evaluating Quorum's Performance under Stressful JSON RPC Loads using Chainhammer

To evaluate Quorum's capacity to process JSON RPC requests, we employed the Chainhammer tool (https://github.com/drandreaskrueger/chainhammer). Chainhammer is an open-source performance benchmarking tool specifically designed for testing blockchain platforms like Quorum. It functions by simulating a high volume of JSON RPC requests, allowing us to gauge Quorum's ability to handle real-world workloads efficiently. Quorum demonstrated a marginally higher processing capacity than Besu under identical test conditions. Quorum achieved an average processing rate of 2,305.82 requests per second when bombarded with 100,000 JSON RPC requests.

6.2.3. Stress Testing Besu TPS with Hyperledger Caliper

To determine the maximum transaction throughput of the Besu blockchain under the setup described in Section 5.1, we employed the Hyperledger Caliper tool. Hyperledger Caliper is an open-source performance benchmarking framework designed to evaluate the performance of blockchain platforms. It provides a comprehensive suite of tools and methodologies for conducting performance tests, enabling developers to assess the scalability, latency, and overall throughput of blockchain networks. We identified our capacity to process transactions per second, supported by the following data: 334 transactions per second for the Besu blockchain, with a standard deviation of 20.

6.2.4. Stress Testing Quorum TPS with Apache JMeter

To evaluate the maximum transaction processing capacity of the Quorum blockchain under the configuration detailed in Section 5.1, we utilized the Apache JMeter tool. Apache JMeter is an open-source load testing tool used for performance testing of various applications, including blockchain platforms. It allows for simulating a high volume of concurrent requests, enabling us to assess Quorum's ability to handle real-world transaction loads efficiently.

The results for analyzing the maximum load capacity that the two blockchains can handle in terms of transactions per second and requests per second are synthesized through graphs and tables [17]. In particular, Figure 2 illustrates four box plots: two for the maximum TPS load that the two blockchains, Quorum and Besu, can handle, and two for the maximum number of requests that the two blockchains can manage.

Table 3 presents statistics regarding the data collected for the two blockchains when subjected to stress tests using the tools listed in Table 1. Among the reported values are the means, standard deviations, quartiles (including Q1, Q2, and Q3), as well as the minimum and maximum values.

7. Discussion

The outcomes of this investigation can be partitioned into two distinct parts.

Specifically, in the initial part, we conducted a simulation involving a mere 20 prosumers. The statistical analysis yielded a t-statistic of -0.1138 and a p-value of 0.9095. In this study, our objective was
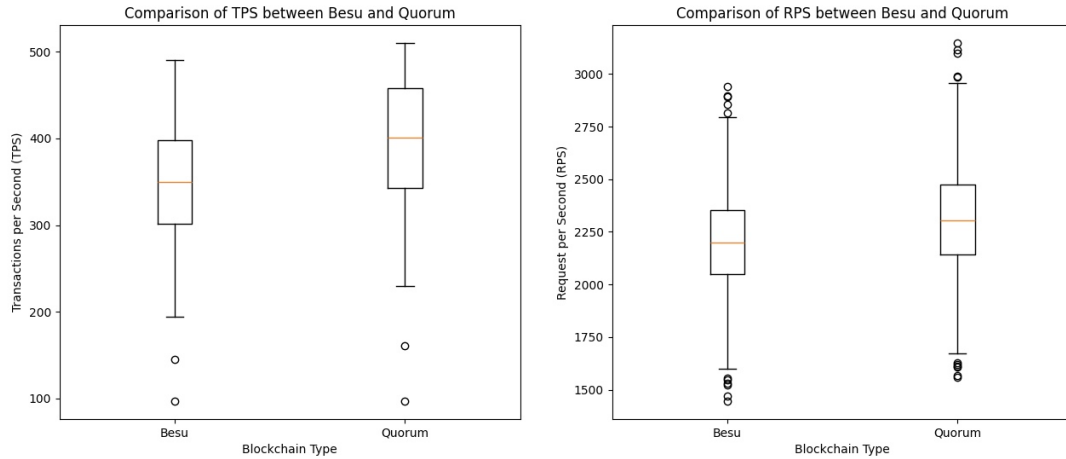
Figure 2: Comparison of Maximum TPS and RPS Load

Table 3
Statistical Analysis under Peak Load Conditions

Test Name	Blockchain Name	Mean	SD	Q0	Q1	Q2	Q3	Q4
Max TPS	Besu	349.75	67.89	97	302	350	398	490
	Quorum	397.56	76.3	97	343	401	458	510
Max JSON RPC	Besu	2198.14	221.1	1446	2049.75	2201	2351.25	2941
	Quorum	2307.46	237.29	1559	2141.75	2305	2473	3147

to compare the transaction volumes between the BESU and Quorum networks. To gauge the significance of any disparities observed, we rigorously executed a statistical examination. However, upon thorough evaluation, we found that the calculated p-value associated with the comparative analysis fell short of meeting the predetermined threshold for statistical significance, typically set at 0.05. This outcome suggests that there exists insufficient evidence to warrant the rejection of the null hypothesis. The null hypothesis posits that there is no discernible distinction in transaction volumes between the BESU and Quorum networks. Consequently, based on the empirical evidence analyzed in this investigation, we cannot assert with confidence the presence of a statistically significant variance in transaction volumes processed by these two blockchain networks. In the scenario we tested, involving energy communities with only 20 simulated users, we didn't observe any differences.

Subsequently, utilizing dedicated simulation tools, we conducted an in-depth analysis of the maximum transactional capacity of Besu and Quorum blockchains, considering the software and hardware configuration employed in our simulation. The analysis revealed significant differences between Besu and Quorum in terms of Transactions per Second (TPS) and Requests per Second (RPS). Specifically, within the hardware and software configuration described in section 5, the Quorum blockchain demonstrated superior performance compared to Besu. The statistical analysis using a t-test yielded a T-Statistic of -47.3566 and a P-Value of 0.0. A P-Value of 0.0 indicates a statistically significant difference between Besu and Quorum's performance in terms of TPS and RPS within the context of our simulation environment. Therefore, we can confidently reject the null hypothesis, which stated that there is no significant difference between the two blockchains. This finding strongly suggests that Quorum outperforms Besu in our specific hardware and software configuration.

8. Conclusion

In this study, we conducted simulations of two distinct scenarios to evaluate the efficacy of our blockchain-based technical solutions in facilitating the management of energy communities [18].

The initial scenario involved a small-scale setting comprising 20 prosumers (entities both producing

and consuming energy) equipped with photovoltaic panels. These prosumers engaged in trading surplus energy and battery storage capacity through blockchain-based tokens. This scenario was designed to gauge the practicality and effectiveness of our platform within a realistic context featuring a limited participant pool.

The subsequent scenario aimed to stress-test our technical solutions, deployed on both Besu and Quorum blockchains. To accomplish this, we utilized open-source tools tailored for simulating the maximum transaction throughput per second (TPS) and the volume of JSON-RPC requests processed for blockchain block data retrieval. Through simulating a significant volume of concurrent users and transactions, we aimed to pinpoint potential bottlenecks and performance constraints within our platform.

Our findings confirm that simulating an energy community scenario involving only 20 producers, trading energy through tokens, shows no performance difference in terms of TPS and RPS between the two blockchains. However, when considering implementation for a much larger-scale environment, the Quorum solution appears to be more efficient in terms of both TPS and RPS compared to Besu. The results of these simulations provided valuable insights into the scalability and performance characteristics of our blockchain-based energy trading platform. This information will be used to optimize the platform and ensure its ability to handle a large number of users and transactions in a real-world deployment.

This analysis helped us to identify potential challenges and opportunities associated with the adoption of our platforms BESU and Quorum. We believe that our findings will be valuable for researchers, developers, and policymakers who are interested in exploring the potential of blockchain technology for energy communities. Future research could delve deeper into the comparative analysis by extending it to different hardware and software configurations to obtain a broader mapping of the relative performances of Besu and Quorum.

9. Acknowledgement

References

[1] R. Saltini, D. Hyland-Wood, Ibft 2.0: A safe and live variation of the ibft blockchain consensus protocol for eventually synchronous networks, ArXiv abs/1909.10194 (2019). URL: https://api.semanticscholar.org/CorpusID:202719257.

[2] D. Ongaro, J. Ousterhout, In search of an understandable consensus algorithm, in: 2014 USENIX Annual Technical Conference (USENIX ATC 14), USENIX Association, Philadelphia, PA, 2014, pp. 305–319. URL: https://www.usenix.org/conference/atc14/technical-sessions/presentation/ongaro.

[3] M. B. Mollah, J. Zhao, D. Niyato, K.-Y. Lam, X. Zhang, A. M. Y. M. Ghias, L. H. Koh, L. Yang, Blockchain for future smart grid: A comprehensive survey, IEEE Internet of Things Journal 8 (2021) 18–43. doi:10.1109/JIOT.2020.2993601.

[4] A. S. Musleh, G. Yao, S. M. Muyeen, Blockchain applications in smart grid–review and frameworks, IEEE Access 7 (2019) 86746–86757. doi:10.1109/ACCESS.2019.2920682.

[5] M. Mehdinejad, H. Shayanfar, B. Mohammadi-Ivatloo, Decentralized blockchain-based peer-to-peer energy-backed token trading for active prosumers, Energy 244 (2022) 122713. URL: https://www.sciencedirect.com/science/article/pii/S0360544221029625. doi:https://doi.org/10.1016/j.energy.2021.122713.

[6] A. A. G. Agung, R. Handayani, Blockchain for smart grid, Journal of King Saud University - Computer and Information Sciences 34 (2022) 666–675. URL: https://www.sciencedirect.com/science/article/pii/S1319157819309000. doi:https://doi.org/10.1016/j.jksuci.2020.01.002.

[7] H. T. Doan, J. Cho, D. Kim, Peer-to-peer energy trading in smart grid through blockchain: A double auction-based game theoretic approach, IEEE Access 9 (2021) 49206–49218. doi:10.1109/ACCESS.2021.3068730.

[8] P. Research, Blockchain technology for the renewable energy sector: A comprehensive study, https://medium.com/paradigm-research/blockchain-technology-for-the-renewable-energy-sector-a-comprehensive-study-836da850d519, 2023.

[9] A. A. Monrat, O. Schelén, K. Andersson, Performance evaluation of permissioned blockchain platforms, 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE) (2020) 1–8. URL: https://api.semanticscholar.org/CorpusID:233434865.

[10] Y. Hao, Y. Li, X. Dong, L. Fang, P. Chen, Performance analysis of consensus algorithm in private blockchain, in: 2018 IEEE Intelligent Vehicles Symposium (IV), 2018, pp. 280–285. doi:10.1109/IVS.2018.8500557.

[11] L. Mostarda, A. Pinna, D. Sestili, R. Tonelli, Performance analysis of a BESU permissioned blockchain, in: L. Barolli (Ed.), Advanced Information Networking and Applications - Proceedings of the 37th International Conference on Advanced Information Networking and Applications (AINA-2023), Juiz de Fora, Brazil, 29-31 March 2023, Volume 3, volume 655 of *Lecture Notes in Networks and Systems*, Springer, 2023, pp. 279–291. URL: https://doi.org/10.1007/978-3-031-28694-0_26. doi:10.1007/978-3-031-28694-0_26.

[12] A. Baliga, I. Subhod, P. Kamat, S. Chatterjee, Performance evaluation of the quorum blockchain platform, CoRR abs/1809.03421 (2018). URL: http://arxiv.org/abs/1809.03421. arXiv:1809.03421.

[13] J. Abdella, K. Shuaib, An architecture for blockchain based peer to peer energy trading, in: M. Alsmirat, Y. Jararweh (Eds.), 2019 6th International Conference on Internet of Things, 2019 6th International Conference on Internet of Things: Systems, Management and Security, IOTSMS 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 412–419. doi:10.1109/IOTSMS48152.2019.8939195, publisher Copyright: © 2019 IEEE.; 6th International Conference on Internet of Things: Systems, Management and Security, IOTSMS 2019 ; Conference date: 22-10-2019 Through 25-10-2019.

[14] G. A. Pierro, R. Tonelli, Can solana be the solution to the blockchain scalability problem?, in: 2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER), IEEE, 2022, pp. 1219–1226.

[15] G. A. Pierro, G. Ibba, R. Tonelli, A study on diem and aptos distributed ledger technology, International Journal of Parallel, Emergent and Distributed Systems (2023) 1–17.

[16] U. A. Yusufoglu, T. H. Lee, T. M. Pletzer, A. Halm, L. J. Koduvelikulathu, C. Comparotto, R. Kopecek, H. Kurz, Simulation of energy production by bifacial modules with revision of ground reflection, Energy Procedia 55 (2014) 389–395.

[17] G. A. Pierro, A. Bergel, R. Tonelli, S. Ducasse, An interdisciplinary model for graphical representation, in: Software Engineering and Formal Methods. SEFM 2020 Collocated Workshops: ASYDE, CIFMA, and CoSim-CPS, Amsterdam, The Netherlands, September 14–15, 2020, Revised Selected Papers 18, Springer, 2021, pp. 147–158.

[18] J. Lowitzsch, C. E. Hoicka, F. J. van Tulder, Renewable energy communities under the 2019 european clean energy package–governance model for the energy clusters of the future?, Renewable and Sustainable Energy Reviews 122 (2020) 109489.