

Blockchain for Smart Homes: Blockchain Low Latency

Reem Jamaan Alzahrani, Fatimah Saad Alzahrani

Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

Email: reemalzahrani4466@gmail.com, fatimahcs53@gmail.com

How to cite this paper: Alzahrani, R.J. and Alzahrani, F.S. (2024) Blockchain for Smart Homes: Blockchain Low Latency. *Journal of Computer and Communications*, 12, 1-15. <https://doi.org/10.4236/jcc.2024.1212001>

Received: October 29, 2024

Accepted: November 30, 2024

Published: December 3, 2024

Copyright © 2024 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution-NonCommercial International License (CC BY-NC 4.0). <http://creativecommons.org/licenses/by-nc/4.0/>



Open Access

Abstract

The inclusion of blockchain in smart homes increases data security and accuracy within home ecosystems but presents latency issues that hinder real-time interactions. This study addresses the important challenge of blockchain latency in smart homes through the development and application of the Blockchain Low Latency (BLL) model using Hyperledger Fabric v2.2. With respect to latency, the BLL model proposes the optimization of the following fundamental blockchain parameters: transmission rate, endorsement policy, batch size, and batch timeout. After conducting hypothesis testing on system parameters, we found that transactions per second (tps) of 30, OutOf (2) endorsement policy, in which any two of five peers endorse a batch size of 10 and batch timeout of 1 s, considerably decrease latency. The BLL model achieved an average latency of 0.39 s, approximately 30 times faster than Ethereum's average latency of 12 s, thereby enhancing the efficiency of blockchain-based smart home applications. The results of this study demonstrate that despite introducing certain latency issues, proper selection of parameters in blockchain configurations can eliminate these latency problems, making blockchain technology more viable for real-time Internet of Things (IoT) applications such as smart homes. Future work involves applying the proposed model to a larger overlay and deploying it in real-world smart home environments using sensor devices, enhancing the given configuration to accommodate a large number of transactions, and adjusting the overlay in line with the complexity of the network. Therefore, this study provides practical recommendations for solving the latency issue in blockchain systems, relates theoretical advancements to real-life applications in IoT environments, and stresses the significance of parameter optimization for maximum effectiveness.

Keywords

Internet of Things (IoT), Blockchain Latency, Hyperledger Fabric, Smart Homes, Latency Optimization, Data Security

1. Introduction

The Internet of Things (IoT) has rapidly expanded across various industries, significantly impacting how we interact with our living environments, particularly through the adoption of smart homes. Smart homes use interconnected devices to automate and optimize activities such as lighting, temperature control, and security systems, thereby enhancing the quality of life for users [1]. Cisco projected that by 2022, there would be approximately 14.7 billion IoT devices in smart homes, underscoring their growing relevance in contemporary lifestyles [2]. However, as the number of IoT devices integrated into daily life increases, concerns about privacy and security have intensified due to the vast amounts of sensitive information transmitted through these interconnected systems.

Traditional security solutions for IoT devices often rely on centralized architectures, which suffer from limitations such as single points of failure, poor fault tolerance, and scalability issues [3]. Blockchain technology, characterized by decentralization, immutability, and peer-to-peer communication models, has emerged as a promising solution to enhance data security and authentication in smart home environments [4]. Smart homes could thus be shielded from attackers by using blockchain to defend against such threats as unauthorized access to smart homes and data breaches. However, incorporating blockchain into IoT solutions and mainly smart homes comes with various problems, and delay is among the most vital ones.

Blockchain network latency refers to the time elapsed between the initiation of a transaction by an IoT device and the incorporation of its details into the blockchain ledger [5]. This latency can impede real-time interactions required in smart homes, as blockchain operations such as transaction validation, consensus mechanisms, and block generation introduce delays [6]. For instance, in the Bitcoin network, transaction confirmation can take up to 10 min [7], which is impractical for time-sensitive smart home applications like unlocking doors or adjusting climate control systems. Even faster platforms like Ethereum have an average latency of 12 s [8], which may still hinder the responsiveness expected in smart home environments.

Existing research has primarily focused on improving blockchain scalability and throughput but has not sufficiently addressed the specific latency requirements of real-time IoT applications in smart homes [9]. The critical knowledge gap lies in the lack of blockchain models optimized for low latency without compromising security and efficiency in smart home contexts. Previous attempts to reduce latency often involve sacrificing decentralization or security, which is unacceptable for applications handling sensitive personal data.

To address this gap, we propose the Blockchain Low Latency (BLL) model, which optimizes key blockchain parameters to significantly reduce latency in blockchain-based smart home applications. Our model focuses on adjusting the transmission rate, endorsement policy, batch size, and batch timeout within the Hyperledger Fabric v2.2 framework. By fine-tuning these parameters, we aim to

achieve a balance between low latency, high throughput, and robust security.

The novelty of our study lies in the systematic approach to parameter optimization for latency reduction in blockchain networks applied to smart homes. Unlike previous studies, we demonstrate that carefully selecting and configuring blockchain parameters can reduce latency to levels suitable for real-time IoT applications without compromising the inherent security advantages of blockchain technology. Our experiments show that the BLL model achieves an average latency of 0.39 s, approximately 30 times faster than Ethereum's average latency. This significant reduction in latency enhances the practicality of deploying blockchain in smart home environments.

This study aimed to reduce blockchain latency by controlling the block validation process, block time, and block size and improve the efficiency of blockchain-based smart homes by increasing transaction throughput. By achieving these objectives, the BLL model ensures that smart home devices can provide the fast, secure, and efficient services expected by users, with latency reduced to as low as 0.39 s—30 times faster than Ethereum.

2. Methods

This study was conducted to develop and implement a Blockchain Low Latency (BLL) model within the Hyperledger Fabric (HLF) platform to address latency issues in blockchain-based smart home environments. The methodology comprises four phases: analysis, design, implementation, and testing. These phases were structured to assess how varying specific blockchain parameters could reduce latency while maintaining the security and efficiency of the system.

Phase 1: Analysis

The analyzing phase was dedicated to identifying the main general requirements that need to be fulfilled to achieve high performance and minimal latency of the smart home applications implemented with the use of the blockchain technology. Furthermore, research has been conducted to determine the functional and non-functional requirements resulting in efficiency in smart home applications based on use of block chain technology. Both functional and non-functional requirements were used during the selection of appropriate components. The following non-functional requirements assumed as relevant were security, data integrity, data accessibility and identity. These requirements were achieved through the use of Transport Layer Security (TLS), a guarantee of transactions through digital signatures, and the use of a large number of nodes that reduce vulnerability points [3]. According to functional requirements, the main processes within the blockchain network are registration, configuration, working, and purchasing. For instance, in the BLL model, new users and devices can be enrolled using the Hyperledger Fabric Certificate Authority (Fabric CA), while several parameters such as the number of transactions, transaction rates, block size, and block timeout are limited, but its low-latency performance is attainable [4].

Phase 2: Design

During the design phase, the BLL model was structured around several com-

ponents: Membership Service Provider (MSP), Fabric CA, client, HLF SDK, endorsing and committing peers, ordering nodes, and chaincode/smart contract. Each of the parts contributed towards reducing the latency time in the transaction execution phase. For example, for identity verification, the MSP was used, and for security and transaction efficiency, endorsing peers was utilized. The BLL model was designed to operate across three main stages.

1) **Registration Stage:** The beauty of Fabric CA is that users and devices are enrolled with the network and issued with credentials and digital certificates.

2) **Transaction Designing Stage:** The client creates transactions by employing the HLF SDK, and the identities of the clients are also checked prior to the transactions being sent to the blockchain network for validation.

3) **Transaction Processing Stage:** The transaction is known to take three phases, which include the execution phase, ordering phase, and validation phase. The endorsement policy, managed by the BLL model, precluded only two confirmations out of five, although this optimized the number of confirmations without considering the security aspect. The out-of-endorsement policy was selected instead of the other policies (AND, OR) to avoid “latency” and bottlenecks due to the necessity of too many confirmations [10].

Phase 3: Implementation

In the implementation phase, the Hyperledger Fabric v2.2.0 platform was deployed on Ubuntu 20.04 on a virtual machine using Intel Core i7 and RAM 16 GB. The operational performance of the BLL model was evaluated by the experiment with a limited dataset of a temperature sensor in the context of IoT devices used in smart homes. The experiments were performed using the Caliper tool, which introduced a variable test latency by varying key parameters such as transmission rate, number of confirmations, batch size, and batch time limit. The experiments were structured as follows.

- **Experiment One:** Determined how changing the transmission rate impacted the latency with 1000 and 2000 transactions. The transmission rate that delivered the best performance was 30; therefore, the average latency recorded was 0.39 s.
- **Experiment Two:** Experiments with different endorsement policies to determine which proved best. Out of (2) policy was the least with a low latency, as it demanded only two confirmations.
- **Experiment Three:** The correlation between batch size, batch timeout, and latency was considered. Latency was found to be reduced by using a batch timeout of 1 s and a batch size of 10 transactions.

Phase 4: Testing

The last step comprised an evaluation of the BLL model to determine whether it corresponded with the goals defined for the project. The BLL model was validated to have offered a massive boost in efficiency, slashing latency to 0.39 s from Ethereum’s typical latency of 12 s [5]. This model successfully dealt with 1000 of the analyzed transactions with a success rate that was evaluated and affirmed to

be at a 100% success rate with the use of the Caliper tool.

These experiments validated the hypothesis that controlling variables such as the transmission rate, endorsement policy, batch size, and batch timeout could effectively reduce latency while preserving blockchain security. The results obtained from the BLL model demonstrated its superior performance compared with other existing solutions when implemented in the Hyperledger Fabric framework, where the consensus can be seamlessly altered based on the network requirement [4].

3. Results

This section presents the results of three experiments conducted to measure and optimize the latency in the Blockchain Layer for a smart home (BLL) model. The primary objective was to identify suitable values for variables affecting latency, such as sending rate, endorsement policy, batch size, and batch timeout, to enhance the performance of the BLL model.

BLL Model Setup

The BLL model was deployed on Hyperledger Fabric v2.2.0, which is compatible with a smart home blockchain layer. All the experiments were performed on a virtual machine with an Ubuntu 20.04 Linux distribution and the required prerequisites already installed. The computational hardware requirements and software environment of the application are listed in **Table 1**.

Table 1. BLL setup requirements.

Environment	Description
Go lang	Go 1.13.8 Linux/amd64
Docker	20.10.13
Docker-compose	1.27.4
curl	7.68.0
Python	v2.7.18
Node.js	v10.19.0
Hyperledger fabric	v2.2.0
CPU	11th Gen Intel Core i7-1165G7
RAM	16 GB
OS	Ubuntu Linux v20.04

Experiments

Three experiments were conducted, each focusing on different variables that influence blockchain latency: the number of transactions, sending rate, endorsement policy, batch timeout, and batch size. To optimize the results, multiple rounds were conducted for each experiment, with the latency measured using

Caliper, a benchmarking tool.

Experiment One: Impact of Sending Rate on Latency

The first experiment aimed to identify the sending rate at which the BLL model could handle transactions without causing a substantial increase in latency. Two cases were examined, both involving a fixed number of transactions while adjusting the sending rate.

Case 1: 1000 Transactions

The sending rate varied from 10 to 100 transactions per second (tps), and the average latency was recorded for each rate.

From **Table 2**, it is evident that increasing the sending rate beyond 30 tps results in higher latency. The optimal sending rate for 1000 transactions was found to be 30 tps, achieving the lowest average latency of 0.39 s.

Table 2. Impact of sending rate on latency for 1000 transactions.

Tx number	Send rate (tps)	Avg latency (s)	Success	Throughput (tps)
1000	10	0.69	1000	10
1000	20	1.42	1000	20
1000	30	0.39	1000	30
1000	40	1.78	1000	36.4
1000	50	2.45	1000	34.4
1000	60	5.11	1000	39.4
1000	70	4.71	1000	47.6
1000	80	7.06	1000	41.9
1000	90	6.30	1000	46.7
1000	100	7.97	1000	44.7

Case 2: 2000 Transactions

This experiment was repeated with 2000 transactions while varying the sending rate.

The results are presented in **Table 3**.

Table 3. Impact of sending rate on latency for 2000 transactions.

Tx number	Send rate (tps)	Avg latency (s)	Success	Throughput (tps)
2000	10	0.73	2000	10
2000	20	0.50	2000	20
2000	30	0.46	2000	29.9

Continued

2000	40	2.72	2000	35.8
2000	50	3.06	2000	43.6
2000	60	6.78	2000	45.6
2000	70	9.12	2000	46.2
2000	80	11.78	2000	45.2
2000	90	25.90	2000	32.8
2000	100	27.74	2000	31.7

Similar to the previous case, the lowest latency occurred at a sending rate of 30 tps for 2000 transactions. Increasing the sending rate beyond this point led to a significant rise in latency.

Conclusion from Experiment One

The BLL model can handle up to 2000 transactions with a sending rate of 30 tps while maintaining low latency. Sending rates higher than 30 tps introduce increased latency, which may affect system performance in real-time smart home applications.

Experiment Two: Effect of Endorsement Policies on Latency

The second experiment focused on determining an endorsement policy that minimizes latency while maintaining network security. Various endorsement policies available in Hyperledger Fabric were tested, manipulating the number of required confirmations as illustrated in **Table 4**.

Table 4. Endorsement policy impact on latency.

Endorsement policy	Required confirmations	Average latency (s)
Solo	1	0.35
OutOf (2)	2	0.39
Majority	3	0.50
All	5	0.70

Endorsement policies requiring fewer confirmations result in lower latency. The OutOf (2) policy achieved a balance between low latency and network security, as it requires any two endorsements from a group of peers, reducing the potential for single points of failure.

Experiment Three: Optimization of Batch Size and Batch Timeout

The third experiment aimed to determine the optimal batch size and batch timeout to minimize latency. The batch size defines the number of transactions in a block, whereas the batch timeout specifies the time between the arrival of the first transaction and the block release.

Case 1: Varying Batch Timeout

With the batch size fixed, the batch timeout was varied from 1 to 5 s as depicted in [Table 5](#).

Table 5. Batch timeout vs. latency.

Batch Timeout (s)	Avg Latency (s)	Success	Throughput (tps)
1	0.39	1000	29.8
2	0.41	1000	29.8
3	0.43	1000	29.8
4	0.51	1000	29.6
5	0.59	1000	29.5

Shorter batch timeouts lead to lower latency. A batch timeout of 1 s resulted in the lowest average latency of 0.39 s.

Case 2: Varying Batch Size

With the batch timeout fixed at 1 s, the batch size was varied from 10 to 50, as presented in [Table 6](#).

Table 6. Batch size impact on latency.

Batch timeout (s)	Batch size	Avg latency (s)	Success	Throughput (tps)
1	10	0.39	1000	29.9
1	15	0.40	1000	29.8
1	20	0.42	1000	29.8
1	25	0.52	1000	29.6
1	30	0.41	1000	29.8
1	35	0.81	1000	29.8
1	40	1.07	1000	29.7
1	45	1.12	1000	29.7
1	50	1.15	1000	28.4

Smaller batch sizes contribute to lower latency. The optimal batch size was found to be 10, resulting in the lowest average latency.

Conclusion from Experiment Three

Decreasing both the batch timeout and batch size reduces latency. A batch timeout of 1 s and a batch size of 10 was identified as the optimal settings for the BLL model, ensuring efficient transaction processing without sacrificing throughput.

Overall Findings

All three experiments provided valuable insights into optimizing the BLL model to minimize latency. A sending rate of 30 tps is optimal for handling up to 2000 transactions while maintaining low latency. The OutOf (2) endorsement policy minimizes latency and enhances network resilience by requiring any two endorsements from a group of peers. Setting the batch timeout to 1 s and the batch size to 10 transactions per block achieved the lowest latency. By implementing these configurations, the BLL model can efficiently process transactions in a smart home blockchain network with minimal latency, thereby improving overall system performance.

BLL Model Testing

After determining the optimal values for the parameters affecting latency, the BLL model was configured accordingly to validate its performance in a real-world scenario. **Figure 1** illustrates the design of the BLL model using optimized parameters.

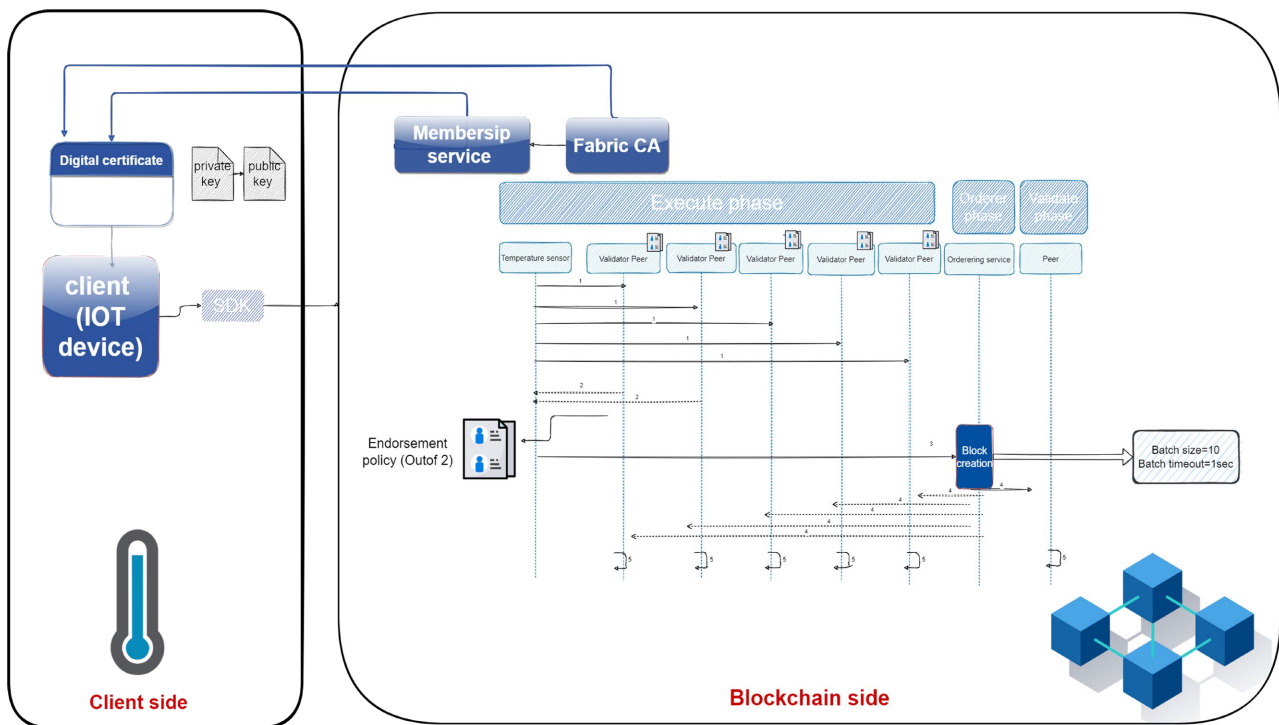


Figure 1. BLL model design.

Figure 1 illustrates the BLL model configured using the optimal parameters derived from the experiments.

Initially, the number of transactions was set to 1000, aligned with the device’s processing capabilities. The sending rate was established at 30 transactions per second (tps) to control the flow of transactions efficiently. The OutOf (2) endorsement policy was implemented in the chaincode and deployed within the network to enhance security while minimizing latency. Finally, the batch size was set to 10,

and the batch timeout was configured to 1 s. The model was tested to ensure that it achieved low latency without compromising the efficiency of smart devices.

An integration test was conducted on the BLL model using these optimized parameters. The results are presented in **Table 7**.

Table 7. Results of BLL model testing.

Tx number	Send rate (tps)	Batch size	Batch timeout (s)	Endorsement policy	Avg latency (s)	Success	Throughput (tps)
1000	30	10	1	OutOf (2)	0.39	1000	29.8

The testing confirmed that the BLL model, configured with the optimized parameters, successfully processed 1000 transactions with an average latency of 0.39 s and a throughput of approximately 29.8 tps.

4. Discussion

This study aimed to address the critical issue of latency in blockchain-based smart home systems by optimizing key parameters within the Hyperledger Fabric framework. Smart homes have been integrating blockchain technology as the use of technology provides security and decentralization. However, latency remains a significant issue that could manifest and compromise the full functionality of real-time smart home systems [11] [12]. Through a series of experiments, we systematically analyzed how varying the sending rate, endorsement policy, batch size, and batch timeout affected the latency and overall performance of the BLL model.

Our experiments revealed that a sending rate of 30 transactions per second (tps) provided the lowest average latency of 0.39 s for both 1000 and 2000 transactions, as depicted in **Table 2** and **Table 3**. Increasing the sending rate beyond 30 tps led to a significant rise in latency due to network congestion and resource limitations. This finding indicates that there is an optimal transaction throughput that balances performance and latency in a blockchain network [13].

We examined various endorsement policies and found that the OutOf (2) policy achieved a balance between low latency and network security, as presented in **Table 4**. This policy requires any two endorsements from a group of peers, reducing the number of confirmations needed compared to stricter policies like Majority or All. By selecting OutOf (2), we reduced the average latency while maintaining a level of security that avoids single points of failure [14].

The experiments demonstrated that decreasing both the batch timeout and batch size reduces latency. Specifically, a batch timeout of 1 s and a batch size of 10 transactions were identified as the optimal settings, as shown in **Table 5** and **Table 6**. Smaller batch sizes and shorter timeouts mean that transactions are processed and committed more quickly, which is crucial for real-time interactions in smart home environments [15].

Our study makes several key contributions. By optimizing the sending rate,

endorsement policy, batch size, and batch timeout, we reduced the average latency to 0.39 s, which is approximately 30 times faster than Ethereum's average latency of 12 s [16]. This significant reduction makes blockchain technology more viable for real-time smart home applications. We provide a systematic approach to parameter tuning in Hyperledger Fabric, demonstrating that careful configuration can enhance performance without compromising security [17]. This framework can be applied to other blockchain-based IoT systems facing similar latency challenges. We also showed that it is possible to reduce latency while maintaining a satisfactory level of security. By selecting the OutOf (2) endorsement policy, we mitigated the risk of single points of failure without incurring the high latency associated with more stringent policies.

We acknowledge that our experiments were conducted on a virtual machine using a limited dataset from a single temperature sensor. While this setup effectively demonstrates the potential of our optimization techniques, it raises questions about scalability and real-world applicability. In real-world smart home environments, numerous devices and sensors operate concurrently. Scaling up the number of peers and transactions could introduce new challenges, such as increased network traffic and resource contention. IoT devices often have limited computational capabilities, so ensuring that the blockchain network can operate efficiently on such devices is critical.

To mitigate these concerns, future work should involve testing with multiple devices, deploying the BLL model in actual smart home settings, and optimizing the implementation for resource-constrained devices. Expanding experiments to include a larger number of IoT devices and sensors would evaluate performance under increased load. Implementing the BLL model in actual smart home environments would allow observation of its behavior under real network conditions. Tailoring the implementation of blockchain to function efficiently on devices with limited processing power and memory is also essential.

Our study focuses on parameter tuning within Hyperledger Fabric to reduce latency, but other approaches exist for latency optimization in blockchain networks, such as alternative consensus algorithms and off-chain solutions. While alternative consensus algorithms like Raft and Practical Byzantine Fault Tolerance (PBFT) can reduce latency, they may introduce complexities or trade-offs in terms of security and fault tolerance [18]. Off-chain mechanisms, such as state channels and sidechains, process transactions outside the main blockchain to reduce on-chain load and latency. However, implementing off-chain solutions can complicate the system architecture and may not be suitable for all types of transactions, especially in smart homes where data integrity and security are paramount.

Our parameter optimization method offers several advantages. Adjusting existing parameters is less complex than integrating new consensus algorithms or off-chain solutions. Our approach is compatible with the existing Hyperledger Fabric infrastructure, facilitating easier adoption. By working within the established framework, we maintain the inherent security features of the platform, preserving

data integrity and confidentiality.

While reducing latency, it is crucial to ensure that security is not compromised, especially in smart home environments handling sensitive personal data. The OutOf (2) endorsement policy reduces the number of confirmations needed but could make the network more susceptible to collusion attacks if both endorsing peers are compromised. To mitigate this risk, we can ensure endorsing peers are distributed and managed by different stakeholders, regularly audit and monitor peer behavior, and implement adaptive endorsement policies that adjust based on transaction type or risk level. Smaller batch sizes, while reducing latency, could increase the frequency of block generation, potentially making the network more vulnerable to certain attacks. Employing robust validation processes and ensuring adequate computational resources can mitigate these risks.

Our approach maintains a balance between performance and security by avoiding single points of failure, maintaining data integrity through immutable transaction recording, and ensuring confidentiality with Hyperledger Fabric's permissioned network model. By providing specific, empirically validated parameter configurations, our study contributes to the ongoing efforts to make blockchain technology practical for real-time IoT applications [18].

Implementing our BLL model in smart home environments can lead to improved user experience through faster transaction processing, enhanced security by maintaining robust security while reducing latency, and facilitated adoption of blockchain technology in IoT applications. The principles demonstrated can be adapted to other permissioned blockchain platforms and IoT applications beyond smart homes [19] [20].

We recognize that our study has limitations, including the experimental scope being conducted in a controlled environment with a limited dataset and resource constraints due to testing on a virtual machine [21]. Future research directions include implementing the BLL model in actual smart home settings with diverse devices, evaluating the model under increased load and network size to assess scalability, and incorporating additional security features like intrusion detection systems and exploring adaptive endorsement policies [22].

By thoroughly discussing our experimental results, analyzing the security implications, comparing our approach with other techniques, and addressing scalability concerns, we provide a comprehensive understanding of our research contributions. This discussion demonstrates how our BLL model advances the field of blockchain applications in smart homes, offering a practical solution to the latency challenges that have hindered the adoption of blockchain in real-time IoT environments.

5. Conclusions

The main objective of this study was to minimize latency as a heuristic measure in blockchain-based smart home settings. By designing a range of experiments and continuously fine-tuning the Hyperledger Fabric platform v2.2.0, we were

able to achieve the desired average latency of 0.39 s on average. This latency rate is significantly lower than that of existing blockchain solutions such as Ethereum, demonstrating the effectiveness of our approach.

To assess the effects of each change, we used the Caliper benchmarking tool on five parameters: the number of transactions, sending rate, endorsement policy, batch size, and batch timeout. Our experiments demonstrated that a sending rate of 30 transactions per second is ideal for maintaining system operation without straining the network. Furthermore, the minimum client latency was recorded with 10 transactions per batch and a batch timeout of 1 s.

Interestingly, we demonstrated in our experiments that the OutOf (2) endorsement policy is the most efficient for the attenuation of latency while ensuring security. This policy mandates that any two endorsements picked at random from the five endorsement peers effectively reduce the number of confirmations needed. Consequently, through the implementation of this policy, we were able to achieve low latency on the network while at the same time maintaining its security. The presented results demonstrate the accuracy of this study and the importance of adjusting the blockchain values in smart home systems. The optimized BLL model offers a practical solution for real-time applications in which low latency is critical and represents enhancements from other blockchain architectures.

Authors and Affiliations

Both authors contributed equally to this manuscript and agreed to be accountable for all aspects of the work.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Morris, M., *et al.* (2013) Smart-Home Technologies to Assist Older People to Live Well at Home.
- [2] Barnett, T., Jain, S., Andra, U. and Khurana, T. (2018) Cisco Visual Networking Index (VNI) Complete Forecast Update, 2017-2022. *Americas/EMEAR Cisco Knowledge Network (CKN) Presentation*, 1, 27.
- [3] Reyna, A., Martín, C., Chen, J., Soler, E. and Díaz, M. (2018) On Blockchain and Its Integration with IoT. Challenges and Opportunities. *Future Generation Computer Systems*, 88, 173-190. <https://doi.org/10.1016/j.future.2018.05.046>
- [4] Zheng, Z., Xie, S., Dai, H., Chen, X. and Wang, H. (2017) An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. 2017 *IEEE International Congress on Big Data (Big Data Congress)*, Honolulu, 25-30 June 2017, 557-564. <https://doi.org/10.1109/bigdatacongress.2017.85>
- [5] Croman, K., Decker, C., Eyal, I., Gencer, A.E., Juels, A., Kosba, A., *et al.* (2016) On Scaling Decentralized Blockchains. In: *Lecture Notes in Computer Science*, Springer, 106-125. https://doi.org/10.1007/978-3-662-53357-4_8
- [6] Bi, W., Yang, H. and Zheng, M. (2018) An Accelerated Method for Message Propagation

in Blockchain Networks.

- [7] Eyal, I., Gencer, A.E., Sirer, E.G. and Van Renesse, R. (2016) Bitcoin-NG: A Scalable Blockchain Protocol. *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*, Santa Clara, 16-18 March 2016, 45-59.
- [8] Pustišek, M. and Kos, A. (2018) Approaches to Front-End IoT Application Development for the Ethereum Blockchain. *Procedia Computer Science*, **129**, 410-419. <https://doi.org/10.1016/j.procs.2018.03.017>
- [9] Kaur, A., Singh, G., Kukreja, V., Sharma, S., Singh, S. and Yoon, B. (2022) Adaptation of IoT with Blockchain in Food Supply Chain Management: An Analysis-Based Review in Development, Benefits and Potential Applications. *Sensors*, **22**, Article 8174. <https://doi.org/10.3390/s22218174>
- [10] Moinet, A., Darties, B. and Baril, J.-L. (2017) Blockchain Based Trust & Authentication for Decentralized Sensor Networks.
- [11] Popoola, O., Rodrigues, M., Marchang, J., Shenfield, A., Ikpehai, A. and Popoola, J. (2024) A Critical Literature Review of Security and Privacy in Smart Home Healthcare Schemes Adopting IoT & Blockchain: Problems, Challenges and Solutions. *Blockchain: Research and Applications*, **5**, Article 100178. <https://doi.org/10.1016/j.bcra.2023.100178>
- [12] Trivedi, C., Rao, U.P., Parmar, K., Bhattacharya, P., Tanwar, S. and Sharma, R. (2023) A Transformative Shift toward Blockchain-Based IoT Environments: Consensus, Smart Contracts, and Future Directions. *Security and Privacy*, **6**, e308. <https://doi.org/10.1002/spy2.308>
- [13] Al-Sumaidae, G., Alkhudary, R., Zilic, Z. and Swidan, A. (2023) Performance Analysis of a Private Blockchain Network Built on Hyperledger Fabric for Healthcare. *Information Processing & Management*, **60**, Article 103160. <https://doi.org/10.1016/j.ipm.2022.103160>
- [14] Kiran, A., Mathivanan, P., Mahdal, M., Sairam, K., Chauhan, D. and Talasila, V. (2023) Enhancing Data Security in IoT Networks with Blockchain-Based Management and Adaptive Clustering Techniques. *Mathematics*, **11**, Article 2073. <https://doi.org/10.3390/math11092073>
- [15] Salmani, M., Ghafouri, S., Sanaee, A., Razavi, K., Mühlhäuser, M., Doyle, J., *et al.* (2023) Reconciling High Accuracy, Cost-Efficiency, and Low Latency of Inference Serving Systems. *Proceedings of the 3rd Workshop on Machine Learning and Systems*, New York, 8 May 2023, 78-86. <https://doi.org/10.1145/3578356.3592578>
- [16] Noor, Q., Multazam, A., Riyadi, S.G., Pratama, R. and Maulana, A. (2024) Large-Scale Ethereum Network Performance Analysis: Case Study and Implications. *Journal of Advanced Technological Innovations*, **1**, 26-34.
- [17] Neiheiser, R., Inacio, G., Rech, L., Montez, C., Matos, M. and Rodrigues, L. (2023) Practical Limitations of Ethereum's Layer-2. *IEEE Access*, **11**, 8651-8662. <https://doi.org/10.1109/access.2023.3237897>
- [18] Shukla, S., Hassan, M.F., Tran, D.C., Akbar, R., *et al.* (2023) Improving Latency in Internet-of-Things and Cloud Computing for Real-Time Data Transmission: A Systematic Literature Review (SLR). *Cluster Computing*, **26**, 2657-2680.
- [19] Wen, Y. and Hsu, C. (2022) A Performance Evaluation of Modular Functions and State Databases for Hyperledger Fabric Blockchain Systems. *The Journal of Supercomputing*, **79**, 2654-2690. <https://doi.org/10.1007/s11227-022-04762-3>
- [20] Zambouri, K., Darbandi, M., Nassr, M., Heidari, A., Navimipour, N.J. and Yalcın, S. (2024) A GSO-Based Multi-Objective Technique for Performance Optimization of

Blockchain-Based Industrial Internet of Things. *International Journal of Communication Systems*, **37**, e5886. <https://doi.org/10.1002/dac.5886>

- [21] Enare Abang, J., Takruri, H., Al-Zaidi, R. and Al-Khalidi, M. (2024) Latency Performance Modelling in Hyperledger Fabric Blockchain: Challenges and Directions with an IoT Perspective. *Internet of Things*, **26**, Article 101217. <https://doi.org/10.1016/j.iot.2024.101217>
- [22] Rao, I.S., Kiah, M.L.M., Hameed, M.M. and Memon, Z.A. (2024) Scalability of Blockchain: A Comprehensive Review and Future Research Direction. *Cluster Computing*, **27**, 5547-5570. <https://doi.org/10.1007/s10586-023-04257-7>