Computing and Informatics, Vol. 43, 2024, 1084–1110, doi: 10.31577/cai_2024_5_1084

SECURE AND EFFICIENT BLOCKCHAIN SCHEME FOR RESOURCE OPTIMIZATION IN INTERNET OF THINGS (IOT) SYSTEMS

Dorcas Dachollom Datiri, Maozhen Li

Brunel University
Kingston Lane, Uxbridge, London, UB8 3PH
United Kingdom
e-mail: {dorcas.datiri, maozhen.li}@brunel.ac.uk

Abstract. The thriving Internet of Things, a paradigm shift from the traditional Internet has brought about great societal improvements such as smart homes, smart cities, smart health, intelligent systems and many more. With these diverse societal improvements come increasing complexities in the areas of system efficiency, privacy, and security. In recent years ample academic and industrial research have delved into resource optimization to the detriment of security, as security features are left to be bolted on at the end of design and developmental processes. This approach leaves the system susceptible to threats and attacks. Consequently, this paper seeks to incorporate security features from the onset, weaving the security feature into the system's design and developmental phase. The proposed model structured in a three-tiered design comprises of concepts of Blockchain, edge computing, clustering techniques and a hybrid algorithm consisting of the static round-robin and the dynamic resource-based algorithms. The composition of the structural layout which considers aspects of the blockchain as a security tight measure for resource optimization in Internet of Things' environment, also incorporates features of edge computing, clustering techniques and the hybrid algorithm as components for resource optimization of the Internet of Things. In addition to the prospective security feature provided by the Hyperledger fabric BC in the proposed model, simulation results illustrate the Hyperledger fabric BC's dexterity in making IoT systems even more efficient, further showing its efficacy over the PSOR2B and the BC-EDSSP.

Keywords: Resource optimization, Internet of Things (IoT), edge computing, clustering, blockchain, security

Mathematics Subject Classification 2010: 68Wxx

1 INTRODUCTION

Internet of Things (IoT) over the years has progressively performed an indispensable part in societal development. IoT devices which are interrelated and connected through the internet, play significant roles in almost every sphere of life, such as, transportation, safety, health, home automation, and different wearable gadgets [1]. The concept of IoT which will dominate the future of Internet communications occurs when devices communicate with other devices on behalf of people [2]. IoT has two major viewpoints: "system view" which divides IoT into blocks, namely Things, Gateways, Network Services, and Cloud Services, and "business view" consisting of Platform, Connectivity, Business Model, and Applications. Regardless of how we describe IoT, there is one common thread – "security is paramount" [3]. The author of [3] states that the biggest challenge facing IoT's security is coming from the very architecture of the current IoT ecosystem, which is based on a centralised model known as the server/client model. While this model has connected computing devices for decades and will continue to support today's IoT networks, it will not, however, be able to respond to the growing needs of the huge IoT ecosystems of tomorrow. With an increase in the use of IoT come an even greater desire for scalability, security, and resource optimization, as data is endlessly transferred between nodes. In the works of [4, 5, 6, 7, 8], the use of edge computing (EC) and or Blockchain (BC) paradigms were epitomized with results suggesting their efficacy in resource optimization and security of the IoT system. The work of [3] suggests that the BC model is a more plausible solution for IoT's threats since BC is viewed as a database that maintains a continuously growing set of data records, [3] further suggests that because of its distributed nature, meaning that there is no master computer holding the entire chain, the BC paradigm further enhances IoT security features. Although [3] suggests BC as a solution for security issues that plague the IoT, their work lacks any form of simulation, thus the efficacy of the BC is not evidently illustrated nor measured. The proposed model uses the best of both worlds by harnessing attributes of both the EC and BC paradigms.

BC, a trustless technology that has been under scrutiny for several years has had several definitions and descriptions: ref. [9] depicts BC as an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way, such that the ledger itself can also be programmed to trigger transactions automatically, [1] further elaborates [9]'s definition by portraying BC as a distributed, decentralized, and immutable digital ledger that records transactions across a global network of computers, making sure that transactions are highly secure. Ref. [10] on the other hand says BC is a distributed database that maintains a growing list of blocks that are chained to each other consequently forming a chain(s) of information that is challenging to corrupt. The authors of [11]

intricate [10]'s definition by stating that BC is an append-only decentralized digital ledger based on cryptography; BC provides a platform to conduct trusted transactions without a third party, in which every fund transfer, every task, and every request has a record on the chain with a digital signature for public verification. The work of [12] on the contrary, labels BC as a data structure, and goes on to say that this data structure makes it possible to create a tamper-proof digital ledger of transactions and share them. The varying definitions put forward by [1, 9, 10, 11, 12] may seem contradictory, but the words data structure, ledger, and database give better sensitivity as to what BC technologies entail. It can therefore be implied that BC is a database because it is a digital ledger that stores information in data structures called blocks, moreso, BC can be inferred to be a data structure as it governs the creation and sharing of transactions that are distributed, decentralised and immutable. It should be noted, however, that not all databases are BCs, but all BCs are databases. Because the BC is based on cryptography, the work of [13] puts forward that BC reduces the Domain Name System (DNS) threat. BC, the immutable distributed ledger allows for registry of devices directly, unlike Domain Name Systems (DNS) [14], thus establishing trust between nodes. BC is of three (3) major forms: permissioned (private), federated (consortium) and permissionless (public). The author of [15] proposes that in a "permissionless" BC such as the one underlying the Bitcoin crypto-currency, anyone can operate a node and participate through spending CPU cycles and demonstrating a "proof-of-work" (PoW), on the other hand, BC in the "permissioned" model controls who participates in validation and in the protocol; these nodes typically have established identities and form a consortium. However, in actual fact, it is the federated BC that has known identities or preselected nodes, consequently forming a consortium. The permissioned BC on the other hand is able to control accessibility of blockchain data via validation and authentication. The key difference between the federated and private BC is that in the federated BC all participating nodes have equal control, whereas in the private BC, just one of the participating nodes has sole control. As oppose the public BC, the federated BC is more performant and does not focus on a crypto-currency. It is worth mentioning that the federated BC, a semi-private BC, allows for better efficiency in the form of better control of energy dissipation, and improved speed, throughput, and security. The proposed model, taking into consideration IoT's functionality and federated BC's attributes, makes use of aspects of the Hyperledger fabric BC, thus ensuring that all participating nodes are authenticated and validated.

The concept of security integrates three fundamental issues: data confidentiality, privacy and trust. According to [16], a possible route to any form of security is the encryption of all functionally encryptable data, accessible only to users with the correct keys, thus protecting its confidentiality against unintentional errors and attacks alike. Encryption, the encoding of information, converts original representation of the information into alternate forms which can only be accessed by authorized parties. With IoT's ever increasing scale, the path of encryption already ladened with key management challenges is a complication worth avoiding. Cryptography

on the other hand, a proper authentication schemes that has the ability to easily identify malicious nodes and illegitimate messages therefore providing security authentication scheme, has shown a great ability to prevent external attacks [17]. The Hyperledger fabric BC's ability to control access to information via validating and authentication of participating parties and its cryptographic techniques, provides the security that encryption would have provided without key management complications tantamount to encryption. Considering the security challenge, IoT device vendors, IoT hardware manufacturers and IoT application developers must include security (authentication, authorization and encryption technologies) in their designing phase, in addition, security testing framework for IoT devices must be introduced. BC has been shown to possess several salient features including security, immutability and privacy and could thus be a useful technology to address the security challenge.

The past decade has witnessed the rapid evolution of BC technologies, which has attracted tremendous interests from both the research communities and industries. Nowadays, BC is envisioned as a powerful backbone/framework for decentralized data processing and data-driven self-organization in flat, open-access networks. In particular, the plausible characteristics of decentralization, immutability, and self-organization are primarily owing to the unique decentralized consensus mechanisms introduced by BC networks [18]. IoT's system view which divides IoT into blocks makes it easy to merge IoT with BC technologies. This conspicuous yet glaring similarity implies a merger that has relevant resource optimization and security prospects. Our previous work [19] shows how the amalgamation of the edge computing (EC) paradigm, particle swarm optimization (PSO) clustering technique and the hybrid algorithm (RR-RB) consisting of the static round robin and dynamic resource based algorithms, labelled PSOR2B, brings about efficient resource optimization strategies with relative security, owing to the EC paradigm. Despite PSOR2B's efficiency, its security feature can be better enhanced. Further incorporating BC into the proposed PSOR2B as depicted in [19], may in actual fact be the best option for maintaining an extensive and transparent record, offering a higher degree of protection for consumer devices, data sharing, and secure payments [20]. The merger will inevitably facilitate faster, securer and smoother transactions. It is to this effect, that this paper seeks to present the following major contributions:

1. A three-tiered framework for a secure and efficient resource optimization mechanism for IoT. The proposed framework further builds on the concepts portrayed in [19, 21] via the incorporation of Hyperledge fabric BC.
2. A mathematical model reflecting the proposed framework. In line with the proposed framework, the model again builds on the mathematical concepts introduced in [19, 21] and delves into secure resource and data management of all nodes concerned.
3. Simulation analysis. The proposed model's effectiveness is scrutinised and also compared with existing models to reflect its efficiency.

The rest of the paper is arranged as follows: Section 2 highlights related works, states the security problem and gives the motivation behind the findings, Section 3 on the other hand presents the proposed model's algorithms description and workflow. Section 4 focuses on the implementation and analysis, and paper ends with concluding statements given in Section 5.

2 RELATED WORK, SECURITY PROBLEM AND MOTIVATION

This section further explores related works and approaches, considering lapses that require solutions. IoT's security problems are expounded with the sole aim of intricating the proposed model's effectiveness. Also, the motivation for this research and approach is given.

2.1 Related Work

BC, a technology that uses community validation to keep synchronized the content of ledgers replicated across multiple users, derives its origins from technologies introduced decades ago [22]. BC over the past years has seen a steady growth in development rate since its recognition as a crypto-currency (Bitcoin). Generally, the term "blockchain networks" can be interpreted from two levels, that is, the "blockchains" which refer to a framework of immutable data organization, and the "blockchain networks" on top of which the approaches of data deployment and maintenance are defined; the two aspects are also considered as the major innovation of BC technologies, for data organization, BC technologies employ a number of off-the-shelf cryptographic techniques [18]. Different views and work have appeared within the past decade with the notion of possibly building a high-performance BC platform for intelligent devices so as to create more value, thus shifting from traditional service providers to data value providers and consumers [23]. A fundamental part of BC is a structure known as the Merkle-Patricia tree. A Merkle-Patricia tree has three types of nodes: a) leaf; b) extension; and c) branch nodes; it is used to efficiently store and retrieve data structures associated with strings [22, 24]. In the BC context, the string is the hash value of the address of an account or transaction, and the data structure to be retrieved is the account/transaction itself. The branch nodes only store the hash value of the list of its child nodes, leaf and extension node, store a key, that is the hash value of the common path shared by all child nodes, and a value. The value stored by extension node is the hash value of the list of child nodes, and the one of the leaf nodes is the hash value of the data that is to be authenticated (e.g., an account or transaction), the use of a hash function to index the addresses provides equal length of the strings [22], and the entries of BC are sequential and time-stamped. The author of [24] imply that a one-way function produces a short bitstring (for example, 512 bits) and depends on every item as well as its placement in the log. The function has mathematical properties that assure that it would be astronomically difficult to produce a different log with the same

output. The output of the function is an abbreviation for the log itself. To add new entries, the function uses its current value and the contents of every new entry to compute a new output. The log maintainer publishes the log and the output value so that independent parties can verify the correspondence.

Because BC works like a ledger ensuring that all nodes perform bookkeeping activities, the work of [23] designed a high performance blockchain platform using technologies such as distributed network architecture, intelligent devices node mapping, the economical model, as well as PBFT-DPOC consensus algorithm to realize the decentralized autonomy of intelligent devices. The approach of [23] focused on system security and stability with the sole aim of improving decentralization in permissioned BCs. Leveraging on the findings that led to [23]'s approach, the proposed three-tiered model goes a step further by considering scalability and resource optimization. The authors of [25] developed a BC-based Public Key Infrastructure (PKI) technology which tackles the underlying problems of PKI technologies by ensuring that certificates on the write BC are trusted; can be used for the issuance and management of self-signed digital certificates across multiple organizations; and can also be used to replace multiple instances of bridge Certificate Authority (CA) connections that use different CAs to issue certificates. In ref. [25]'s approach, the privacy aware permission BC consisting of registration BC (RBC), certificate BC (CBC) and user, uses the multiple signature scheme as well as the Concurrent Byzantine Fault Tolerance (CBFT) algorithm, a dual-chain architecture, composing of Account BC (ABC) and transaction BC (TBC) to ensure data is easily stored and processed with minimal data tampering. In spite of [25]'s positive outputs which have the characteristics of anonymity, conditional traceability, and realizing the separation of user identity authentication and legitimate authorization, their work is domain-specific focusing on publishing scheme of anonymous digital certificate and not applicable to the varying and dynamic IoT networks.

The works of [18, 22, 26, 27] position that BC enables shared access to information which is broadcasted across a network based on the trust of its participants; the BC is saddled with the core task of ensuring that the trustless nodes in the network reach the agreement upon a single tamper-proof record of transactions without an expensive "mining" activity as in the bitcoin application. This is achievable via the use of a number of unconventional protocols, such as Bitcoin's simplified payment verification (SPV), the Ethereum light client, lightweight protocol, aggregation protocol and so on. These protocols are used to enhance, reduce and manipulate all necessary data communications. There are several types of BC platforms available; Table 1 gives a summarised comparison of six (6) BC platforms and their relevant properties, with an indication of their relative impact on quality in an IoT context.

Key:

1. PoW = Proof of Work,
2. BFT = Byzantine Fault Tolerance,
3. PoC = Proof of Correctness,

4. RPCA = Ripple Protocol Consensus Algorithm,
5. RRs = Round-Robin Schedule,
6. * = Least favourable; ** = Less favourable; *** = More favourable; **** = Most favourable.

Hyperledger fabric uses the Practical Byzantine Fault Tolerance (PBFT) consensus and is made up of four (4) components: peer – stores all transaction; orderer – orders transactions; CA – manages users' certificates; and client – application that interacts with the Hyperledger fabric network. Compared to the classical Byzantine consensus protocols allowing very limited network scalability, most of the existing consensus protocols in open-access BC networks (e.g., Bitcoin) guarantee the better network scalability at the cost of limited processing throughput. Also, to achieve decentralized consensus among poorly synchronized, trustless nodes, a number of these protocols incur huge consumption of physical resources such as computing power. Moreover, to ensure a high probability of consensus finality, the protocols may also impose high latency for transaction confirmation [18]. The ephemeral nature of users' pseudonymous identities in Bitcoin played a key role in its early success. However, years of intense scrutiny by privacy researchers has brought to bear an arsenal of powerful heuristics which attackers can effectively link disparate Bitcoin transactions to a common user and, in many cases, to that user's real-world identity [28]. Ref. [28] goes on to say that ultimately, instead of providing the bastion of privacy for financial transactions that its early adopters envisioned, Bitcoin and its altcoin brethren are in many ways less private than traditional banking, where government regulations mandate basic privacy protections. In an attempt to address this situation, the cryptography and privacy research communities have proposed and implemented several protocols aiming to improve BC privacy. These protocols all try to decouple users' pseudonymous identities from the specific transactions they make, thereby frustrating attempts to link transacting parties based on data that appears in the BC. However, none of the proposed protocols attempt to hide the identities of users from network-level adversaries as the users publish or retrieve data from the BC. Instead, the proposed protocols "outsource" this crucial step, relying on an external anonymous communications network such as Tor. However, running complex protocols over general-purpose, low-latency anonymity networks such as Tor is fraught with risks and can expose users to subtle-yet-devastating attacks, thereby undermining the privacy guarantees of the entire BC system [28]. The paper produced by [23] tested the transaction throughput and system delay of the intelligent device BC and compared it with the performance of public BC such as Bitcoin and Ethereum with the main purpose of designing a high-performance BC platform for intelligent devices. The platform achieved efficient connection of intelligent devices through the node-to-node mapping mechanism of intelligent devices. At the same time, they designed a BC consensus algorithm for intelligent devices, which provides higher consensus efficiency while guaranteeing decentralization, providing higher efficiency, this enabled making all the relevant parties of the intelligent devices obtain higher efficiency and benefits to achieve a result of multi-

Type	Name	Scalability	Consensus		Network Size	Anonymity	Fee	Block Size	Smart Contract	Security	Language	Open Source	Uses/ Applicability
Permissionless	Bitcon	*	****	PoW	****	***	**	*	*	***	C++	Yes	Financial transaction
	Ethereum	***	****	PoW	****	***	***	****	****	***	C++ and Python and Java and Golang	Yes	Application developers
Federated	Hyperledger Fabric	****	****	BFT	N/A	****	N/A	****	****	****	C++ and Python and Java	Yes	Application developers
	Ripple	***	****	PoC RPCA	****	****	N/A	****	****	****	C and C++ and Java	Yes	Financial transactions
Private	Multichain	****	****	RRS	N/A	****	N/A	****	*	****	C++ and Python and Java	Yes	Application developers
	Eris	***	****	PoW	N/A	****	N/A	****	****	****	C and C++	Yes	Application developers

Table 1. Blockchain based platform comparison

win. However, with this approach comes an increased number of blockchain-links (ψ) generated for each ledger operation.

Given the practicality of the Hyperledger fabric BC, as depicted in Table 1, the considerations and analysis made by [20, 23, 29, 28], as well as Hyperledger fabric's potential to work well in the proposed model makes it the platform of choice. Aspects of the Hyperledger fabric incorporated into the PSOR2B of [19] ensure optimal security, moreso, Hyperledger fabric's limited scalabilty trait is dampened via PSOR2B's three-tiered structure, credit of the EC paradigm and PSO clustering technique. As purported in [19] and [21], the merger of these algorithms as well as the RR-RB algorithm enables efficient resource allocation. One of the major challenges associated with IoT resource allocation is the service-level agreement (SLA) [4]. SLA, a contract that identifies the Quality of Service (QoS) between a service provider and a user must be provided by IoT at all instances of resource allocation to bring about resource optimization. [30] recommend a BC-based cloud SLA violation monitoring and auditing model as a solution to existing SLA monitoring solutions which lack multi-party trust, have weak audit ability, or have privacy issue. Leveraging on Edge computing's use of the Service Level Agreement (SLA) as a commitment between a service provider and a client [4, 31], concepts of BC's effective trustworthiness and QoS more or less buttress the SLA's structure thus improving trustworthiness by decentralising and making data immutable. The author of [26] states that clustering may involve a centralized entity, which is against the principles of the BC. However, with a distributed phenomenon over the selection of Cluster Heads (CHs), the principles of BC remain intact and the network can be managed efficiently. Distributed clustering helps to lower the number of updates for ledgers as well as reduces the number of ψ. Moreover, the slot-wise transactions offer better control over the operations of the entire network. In general, the major factors in the proposed approach are about the selection of CHs, decision to transmit (when and how), the number of permissible BC queries, and location-based ledger-offloading. Satisfaction of all these issues through stochastic volatility helps to sustain the network for longer duration. The proposed model focalising on clustering techniques' distributed nature, lowers the number of ψ required for efficiency.

2.2 Security Problem

The IoT environment equates ample data generation and manipulation; because so much data is required in managing of an IoT network, data management and resource optimization are paramount. Data storage, processing, and analytics are fundamental requirements necessary to enrich the raw IoT data and transform them into useful information [32]. With an increase in IoT usage, there is an increasing desire for efficiency which has an effect on the QoS and consequently users' Quality of Experience (QoE). Resource optimization over the years has increasingly become an area of key interest, studies have produced several approaches, however, these approaches have very little concern on the security of the IoT and leave security

solutions to be added at the end of developmental processes. The largest challenge facing commercial scale adoption of BC technology is its current inability to meet the requirements of multiple, diverse, and complex business scenarios. Therefore, the "one size fits all" BC approach that is currently utilized by other chains is not viable if BC is to succeed in the future [33]. The larger the number of device connections the slower the network [34]. The computational problems give rise to further errors such as latency and traffic overhead. Considering that IoT nodes have squat processing capabilities and data buffer that cannot practice ciphering computation [35], IoT systems are prone to security issues. Despite IoT's many advantages, its centralised topology is tantamount to underlying security issues that make it susceptible to security threats. The assimilation of BC into the IoT environment can advance security, owing to its application of practices in accordance with the mainstream contributors, and authenticating communications to avoid deceiving and data stealing. Threat components often come from third-party nodes or services which present limited transparency about the security and privacy features they offer. In these environments it is difficult to analyse and ensure the overall system security and privacy levels that can be served to end users. Introducing BC as an effective medium for tackling security and privacy issues in IoT would enable the sidelining of such third-party nodes when they do not conform to set protocol. The IoT-BC amalgamation would be a minor change as its distributed ledgers' mechanism would blend in well with the physically distributed devices of the IoT network and decentralising IoT's topology will further aid BC's effectiveness.

The work of [11] describes the "security IoT-BC framework" which can provide great assurance for IoT data and various functionalities as well as desirable scalability including authentication, decentralized payment, and so on. This framework consisting of four (4) layers: Physical Layer; Communication Layer; Database Layer; and Interface Layer, groups all necessary activities associated with the IoT-BC amalgamation, consequently enabling better proposed solutions. The Physical Layer is synonymous to the smart devices equipped with sensors and actuators; the Communication Layer equates to the networks and protocols, that is different communication mechanisms that IoT uses for access to the system and data exchange, such as WiFi, 4G, and Ethernet; Database Layer on the other hand handles the data; finally, the Interface Layer contains applications that communicate with each other to make beneficial decisions collaboratively. The proposed model, taking into consideration this security framework, and in correspondence to the four layers, is applicable for all types of physical devices, all IoT networks, uses the federated distributed ledger – Hyperledger fabric and is not restricted to any application.

2.3 Motivation

Whilst BC becomes more and more widely used as a security mechanisms, especially for handling large scale transactions, its transaction processing capacity faces tremendous pressure when using sequential processing, which results in the bottleneck of network performance [33]. Despite BC's pros, the work of [33] describes how

current BC systems are not yet capable or efficient enough to function as a versatile operating system supporting multiple applications, furthermore, BC technologies face multiple challenges for improving capacity, sometimes at the expense of transaction efficiency. Existing BCs: are not scalable, as the performance of one single node/mining machine determines the performance of the whole system; do not segregate resources for different smart contracts, which causes interference between smart contract executions; do not have pre-defined consensus protocol to adopt updates or adapt to new technology. The work of [32] reveals that overall, BC as a technology has the potential to change the way transactions are conducted in everyday life, however, they also claim that the current BC has a possibility of a 51 % attack. In a 51 % attack a single entity would have full control of the majority of the network's mining hash-rate and would be able to manipulate BC. BC suffers from technical limitations and challenges. Recently research and study show anonymity as an advantage of BC since threats concerning anonymity are easily handled, however, scalability still poses more concerns especially in relation to security challenges [32]. Additionally, privacy is still an issue, and for an apt solution, one has to consider the attributes of the intelligent devices used. Although many researchers advocate using anonymous communications networks, such as Tor, to ensure access privacy, it has been argued in some instances that the communications network Tor, an area where research has dwelt in the past few years, is in fact not the best option [28], considering that hackers can still trace the user. Ref. [28] presents an alternate approach, showing the need for mechanisms through which non-anonymous users can

1. publish transactions that cannot be linked to their network addresses or to their other transactions, and
2. fetch details of specific transactions without revealing which transactions they seek.

Because of BC's attributes, potential applications for all organizations abound, however, prevailing studies tend to focus on BC functionality as a target application for payment settlement through supply-chain management and the likes.

Anonymity, scalability, bottleneck, data integrity and security attributes set a lot of interesting challenges and questions that need to be solved and assessed with high quality research. It is with these aforementioned lapses in mind that the proposed model came to be. To answer the question what computational and algorithmic theories are suitable, in practice, for management and optimisation of resources and security properties for Internet of Things? A framework appropriate for modelling and reasoning about resource optimization and security requirements in IoT and a mechanism for managing security requirements and optimising the selection of appropriate security controls will be presented. The proposed three-tiered topology incorporating the positives of the EC, RR-RB, PSO and BC concepts to produce a model that is scalable, secure, avoids bottleneck and fosters anonymity is not limited to just payments but is relatable to all IoT activities. The approach of

this paper is to introduce concepts of security from the onset by incorporating BC technologies to the PSOR2B model of [19]. The Hyperledger fabric BC will eliminate third party intrusions, and, alongside the EC paradigm, PSO clustering technique and adaptive RR-RB algorithm bring about resource optimization in IoT systems. Extracting and merging the positive aspects of EC, PSO, RR-RB and BC into a new concept will bring about a robust and efficient IoT that will enable a secure and efficient resource optimization mechanism.

3 PROPOSED SOLUTION

This section presents the proposed solution by giving an overview of all the algorithms incorporated in the model. Emphasis is made on the Hyperledger fabric BC's role in further optimizing and providing security features. The section additionally describes the proposed model's functionality, showing how the blocks are created, propagated and offloaded.

3.1 Model Algorithm and Architecture

The proposed model as depicted in Figure 1 consists of the models presented in [19] and [21], as well as the Hyperledger fabric BC technology. The proposed topology, labelled PSOR2B-BC, incorporates aspects of the RR-RB algorithm, EC paradigm, PSO clustering technique and BC technology to form a three-tiered topology which allows for a secure and efficient resource optimized system. PSOR2B-BC's 3 major layers are:

1. Edge node layer – first layer: This layer is made up of clusters of IoT devices that have varying computational, bandwidth, and storage capacity. The nodes at this layer are riddled with energy dissipation problems.
2. Dew layer – second layer: This layer contains the CHs and edge servers. The CHs communicate with edge nodes, each other, and servers, they also propagate and offload BCs amongst themselves and onto the servers – mostly edge servers. The edge servers likewise communicate with CHs as well as cloud servers, and in some cases, edge nodes. They also, like the CHs, offload blocks onto cloud servers.
3. Cloud layer – third layer: This layer constituting cloud servers, acts partly as repositories. The servers at this layer communicate with each other, edge servers and in some cases the CHs and edge nodes. They also, like the CH consortium, propagate blocks of the BC amongst themselves, that is, participating servers. Information can also be downloaded from the cloud servers.

The EC paradigm which is a distributed computing architecture that brings about more decentralization of data aims at enabling request processing closest to end users or system's data. EC plays a major role in structuring the nodes of the

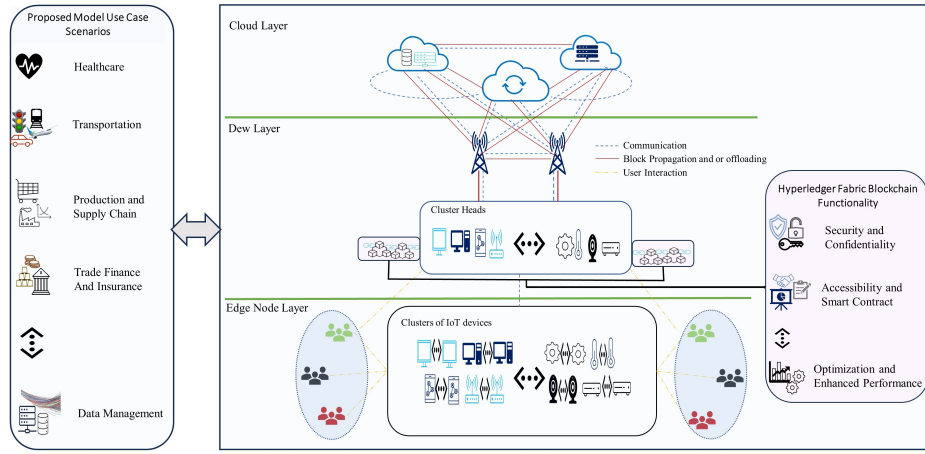

Figure 1. Model design

proposed model into the three layers. The PSO clustering technique further buttresses the EC's structuring by clustering the edge nodes based on the criterion of proximity to edge servers, with the CH usually being the node closest to edge servers with relatively good processing, storage and power capacity. The RR-RB algorithm on the other hand works efficiently to bring about adequate load balancing; where resources are appropriately scheduled and allocated to bring about resource optimization. The PSO also lends a hand in scheduling of resources. Finally, hinging on the federated BC characteristics, some properties of the Hyperledger BC are incorporated into the CH nodes and participating servers, thus adding another layer of security as confidential information can only be accessed by permitted participants within the network. The BC paradigm in contrast to existing SLA of EC, guarantees all records are immutable, consequently providing trust and privacy. Considering that existing BC solutions are prone to long daunting chains as they assume that the BC is maintained by all nodes, the PSOR2B-BC circumvents this problem by ensuring that the impractical construct, especially for macroscale IoT, requiring that each node keeps an exact copy of the BC to guarantee consistency can be manoeuvred. The number of nodes required to maintain the BC in the PSOR2B-BC are drastically reduced by the non-implementation of BC technologies at the edge node layer; BC implementation is at the dew and cloud layers, and even then only participating nodes are indulged. The work of [29] extensively discusses scalability issues associated with BC and IoT, and in their analytical comparisons of scalability solutions, they classed solutions into three approaches: layer zero approach – concerned with the dissemination of information, where the proposed solution focuses on customizing the propagation protocol of information; Layer One – approaches within the BC where the proposed solutions focus on tackling the problem by changing the structure of blocks and consensus algorithms; and Layer Two – approaches of the

BC, where the proposed solutions tackle the problem by executing some complex computational tasks off the BC platform. Based on their findings and assertions, the intended PSOR2B-BC model for scalability purposes and overall system efficiency indulges aspects of all three scalability solution approaches. The PSOR2B-BC limiting BC's functionality to majorly the dew layer customizes the propagation protocol, slightly changes the structure of the blocks and consensus algorithm, and, executes complex computational tasks off the BC platform.

The authors of [36, 37] illustrate the importance of BC in accessibility, identity verification, and storage techniques for data management and communication/data transmission procedures. Ref. [38] on the other hand suggests that in terms of data management, clustering, an unsupervised classification algorithm that aims at classifying data into several disjoint subset, pending on data features has been widely used for text classification, biometric feature recognition, and image segmentation. Although the authors of [39] focus on Internet of Vehicles, the highlighted significance of the EC paradigm establishes how the use of appropriate algorithms can bring about enhanced performance. Furthermore, ref. [17]'s approach on securing communications in vehicle ad hoc networks via a multi-tier trust-based security mechanism highlights the importance of data integrity and breaking-down a problem into smaller chunks to bring about a robust solution. Consequently, the EC, RR-RB, PSO, and Hyperledger BC merger, an amalgamation of relevant algorithms can be purported to produce a full-flavoured solution for the current IoT problems.

The inclusion of the Hyperledger fabric BC to the model's design depicted in our previous works [19, 21] brings about the much-needed security functionality. The ability of the Hyperledger fabric BC to inculcate better security, confidentiality, and accessibility features via the use of smart contracts as well as the Hyperledger fabric BC's ability to further improve optimization, bringing about the model's enhanced performance, implies that the proposed model's usability is versatile. Consequently, the proposed model can be suggested to be a general framework applicable to all IoT contexts. Incorporating the BC's consensus will also ensure efficiency, this is owing to BC technologies' ability to eliminate time consuming processes prone to human error oft at times requiring third-party mediation. The added efficiency that comes with the inclusion of BC is reflected in the efficacy of server rates. The information contained in the CHs resultant of the incorporated BC will provide the best cause of action for each given request. This is made possible by the CH's ability to keep track of servers reachable by edge nodes within its cluster, as well as the most relevant resource for any given request. It is therefore worth mentioning that the BC further enhances system performance consequently strengthening resource optimization in IoT. For storing varying amounts of data, the PSOR2B-BC's use of the Hyperledger BC paradigm implies the use of hash functions and the Merkle tree concept in addition to the mathematical concepts depicted in our previous works [19, 21]. These aforementioned concepts show explicitly how the hybrid algorithm as well as the PSOR2B functions. The Hyperledger BC builds the Merkle tree by having a hash function H and a set of data D where $D = \{d_1, d_2, \ldots, d_n\}$. The tree leaves are

represented as D: $H(d_1)$, $H(d_2)$, ..., $H(d_n)$, the generated blocks B are defined as a vector of entries, $B = \{e_i, e_{i+1}, \ldots, e_j\}$, and the validation protocol $V : N \times B \mapsto \{True, False\}$ and $(n, e) \mapsto V(n, e)$, where N is the set of nodes. For an established smart contract $V(n, e)$ must be true.

In relation to the hybrid load balancing algorithm of [21] and the PSOR2B depicted in [19], aside adding the much-needed security features, the proposed PSOR2B-BC model goes a step further in improving resource optimization in IoT systems. This is achieved by the inculcation of Algorithm 1, as depicted in Table 2. Table 2 presents the algorithm showing how the proposed PSOR2B-BC creates the optimal paths that are recorded in the CHs as blocks, these generated blocks are shared with all participating nodes to ensure immutability and transparency, consequently enhancing security, efficiency, speed and traceability.

Algorithm 1: CH Optimal Path Block Creation
Input: CH: Cluster Head; R: Request; E: Edge node; BC: Blockchain
Output: OP: Optimal Path
Data:
1. **If** request is sent from authorized E to CH
2. CH Checks for OP from BC records to process R
3. **elif** BC contains only genesis block
4. CH creats new OP
5. CH records new OP as a new block
6. **else** R is rejected

Table 2. Algorithm 1

Contingent upon receiving a request from the edge nodes, the CH first validates the edge node's authenticity to eliminate unwanted interference. If the edge node fails authentication and validation checks, it is rejected, however, if the edge node passes authentication and validation, the CH then checks for existing optimal paths for the furtherance of the request processing. As soon as an optimal path is detected, the CH can efficiently schedule and allocate resources for processing. For efficient scheduling and allocation of resources, consequent of the optimal path record held in the CH, the size of the request coming from the edge node and the capacity of the resource (servers) are taken into consideration. It is worth mentioning that the optimal path which is created based on processing time records returned to the CHs by edge nodes upon completion of request processing, is the shortest and most effective route for processing of requests. Processing time is the duration of time for producing an output, minus the wait time. In the case where optimal path records do not exist, the CH records a new optimal path upon completion of request processing. In the event where a newer optimal path is created, the existing optimal path present in the CH is deleted, and the new optimal path is stored in its stead. All records are stored and or updated as blocks and propagated to participating CHs. Efficient scheduling and allocation mechanisms provided by the PSOR2B-BC drastically reduces wait time, thereby promoting overall system efficiency.

3.2 Model Workflow

In the traditional BC networks, all edge nodes have the opportunity to mine and store the blocks [40], in the proposed PSOR2B-BC, however, the edge nodes do not make use of BC and have no need to store blocks. By so doing, the edge nodes are not inundated with storing blocks and so have less storage problems that may lead to inefficiency. Although the edge nodes are not legit miners, they somewhat have the attributes of miners, considering that they provide information regarding their requests, and, requests' processing time to the CHs. The CH act as the verifiers and process information provided by the edge nodes (miners). When the designated CH has limited storage because of excess BC data, redundant data found on the BC is offloaded to the cloud. In the odd case where the edge nodes have limited storage because of BC data, offloading information such as the identifier of the offloading node, blocks' identifier and offloading time will be sent to the CH. The cloud receiving the offloading BC will feed back a completed message after reaching a consensus and after the propagation of the offloaded data through the cloud P2P network. In the PSOR2B-BC, the CHs are the peers that consist of orderer and CA, therefore, the CHs enable creation of blocks of transactions, as well as managing queries or invocations based on permission granted; the client is any of the edge nodes initiating a request that requires the aid of the CH at any given time. The edge node can interact with CH based on its permissions, roles and attributes regulated by the CA. Figure 2 further elaborates what happens at the first two (2) layers – that is the edge node layer and the dew layer, giving an illustrations of block generation and propagation in the context of request transactions governed by the BC. Figure 3 on the other hand expatiates the transactions between the CHs and Servers of the dew and cloud layer.

Figure 2 illustrates the request transaction workflow in the PSOR2B-BC model. Prior to the initiation of requests by the edge nodes, after the creation of clusters, the CH endorses all edge nodes within its cluster, verifying the edge node's identity and authorization. Thus, when the edge node sends a request, its endorsement is attached to the request. Once the request is sent to the CH and the verification check passes, the request is processed, and the edge node is notified. The orderer, a service incorporated in the CH and one of the most important components of the Hyperledger fabric, enables the generation of new blocks of transaction. The generated blocks are signed with the orderer's certificate before finally broadcasting the generated block to all peers via relevant channels. In the case where verification checks fail, the request is rejected without creating or distribution of blocks. The CHs perform a versioning check known as the multi-version concurrency control (MVCC) after every broadcast to validate that the blocks are up to date, thereby eliminating information corruption and redundancy.

Figure 3 demonstrates how the propagation and offloading of blocks in the dew and cloud layer works. At the dew layer, once the new blocks have been created and propagated amongst CHs, checks for redundancy are carried out and redundant blocks are offloaded to the edge servers. The edge computing servers which are

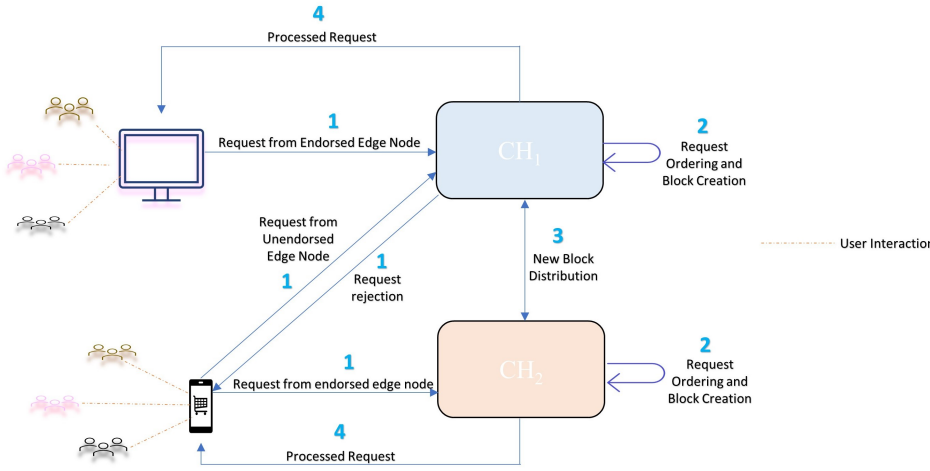

Figure 2. Request transaction workflow

equally running BC technologies, upon reception of the offloaded blocks, disseminate blocks amongst their consortium to ensure the chain is up to date. Similarly, the edge servers offload redundant blocks to the cloud servers found in the cloud layer, where further redundant BC blocks are stored.

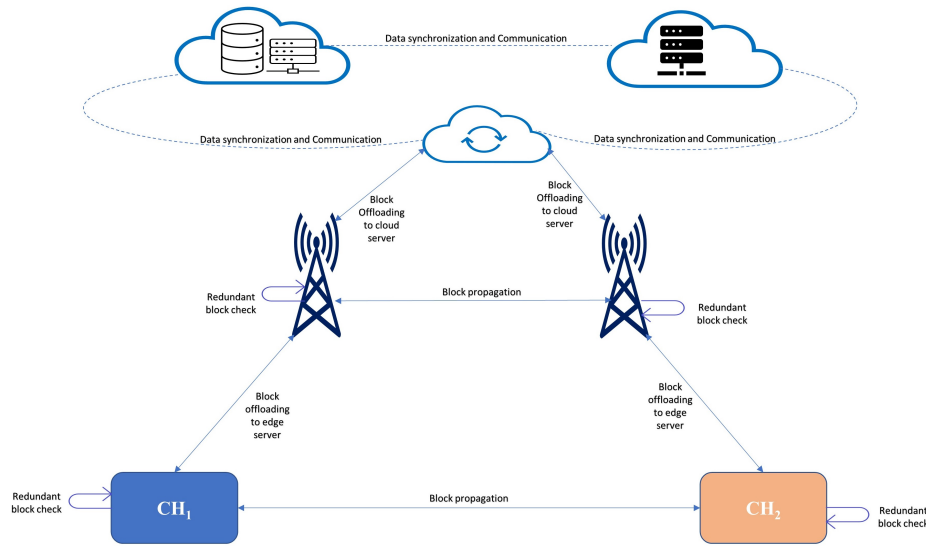

Figure 3. Block propagation and offloading

4 IMPLEMENTATION AND ANALYSIS

This section delivers the implementation procedure for the proposed PSOR2B-BC model. Furthermore, the results generated are analysed to ascertain the efficiency of the implemented PSOR2B-BC.

4.1 Implementation

The simulation setup is made up of five (5) servers/resource and one to one hundred (1–100) nodes. This simulation provides a scarcity of resources akin to real life events. Moreso, the simulation setup mimics the varied capacities the edge nodes of the corporeal environment have. The parameters that governed the implementation as depicted in Table 3 are request size and number of requests, both generated by the edge nodes; resource throughput as produced by server/resource; and the Cartesian distance of resource/server to edge node/request as indicated by both the edge nodes and server/resource. These parameters govern and regulate the generation of the results presented, thus making analysis measurable.

Parameter	**Parameter Specification**	**Systéme International (SI) Unit**
Distance	Cartesian distance of servers from edge nodes	Meter (m)
Time	Turnaround time	Mili-second (ms)
Data transmission rate	Serve/resource rate (Request per second (RPS) Server/resource throughput/transfer rate	bits per second (bps) bits per second (bps)
size	Length of chain Number of requests Request size	Blocks Number (n) Kilo-bits (Kb)

Table 3. Simulation parameters

The simulation setup uses Python programming language and the Anaconda framework to present outputs that exemplify BC's worth in resource optimization and security provisioning. At the implementation phase, there was a block creator, that is a solver/class acting as a regulator. For efficiency, the regulator consistently contains updated information in every block regarding the performance level of each resource/server. Records of the performance level of resources provided by the regulator correspond to information returned by edge nodes that made use of the resources; a reflection of the proficiency of services provided to the edge nodes. The lower and upper limit of each server is recorded on the BC, this matches the structure of a typical BC which has the inner and outer hash. The data stored in the lower and upper limit corresponds to the worst and best case of each resource, thereby

enabling informed decision making as to which resource best fits any given request from the edge node of concern at any given time. Also, on the chain, an efficient path is recorded, this information is immutable, thus deterring any interference from third-party components with unsatisfactory transparency. Because the BC is robust, the redundancy aspect of distributed systems is tackled and information is made available irrespective of catastrophic event at any node, furthermore, by using the BC, the overall system's security and privacy level is better improved. As oppose the PSOR2B which may be susceptible to storage issues as well as a rising potential of over utilization of the CH that may consequently lead to nodal failure, addition of the BC further regulates the storage space required. PSOR2B-BC ensures the size of data stored on the CH is regulated by offloading excess and redundant information found on the chain onto the edge servers and subsequently the cloud, additionally, the Hyperledger fabric BC used in the PSOR2B-BC as oppose public BCs such as Ethereum [20], improves: efficiency by regulating the number of BC participating CH nodes; and security by holding immutable records of data, in form of trustworthy blocks in the participating CH nodes. Finally, Hyperledger fabric BC as oppose the Ethereum BC is not characterised by high energy consumption, thus its incorporation helps regulate energy dissipation by limiting the processing requirements of the CH.

4.2 Performance Analysis

The performance analysis of the PSOR2B-BC is evaluated based on throughput metrics of the proposed PSOR2B-BC versus the PSOR2B of [19]. In addition, considering that one of Hyperledger fabric's drawbacks is its lack of proven use cases, comparisons, also based on the throughput metrics, are made against the popular ethereum public BC of [20].

Figure 4 is a visual summary displaying the effects of the Hyperledger fabric BC on the number of requests servers process per second (server rates). The genesis block server rates refer to the server state before any form of BC intervention, akin to that of the PSOR2B of ref. [19]. Typically, on the BC, genesis block is the first block, that is, the performance obtainable without reference to the BC storage. At this stage, just the algorithms found in the PSOR2B are functional. The BC found in the CH acts as a database/ledger that stores the optimal path, thus PSOR2B-BC's ability to appropriately schedule and allocate resources.

From Figure 4, it can be seen that the genesis block's server rates are lower than the server rates of PSOR2B-BC. PSOR2B-BC's improved server rates equates better system efficiency, consequently improving QoS; the better the QoS, the better the users' QoE. The latest block at any given time is tagged the current block; the current block located in the CH, constituting of records of optimal paths for appropriate scheduling and allocation of resources ensures improved measures for resource optimization in IoT systems. The current block server rates consisting of the algorithms of the PSOR2B in conjunction with Hyperledger fabric BC bring about the optimal performance obtainable at the latest block server rates. The

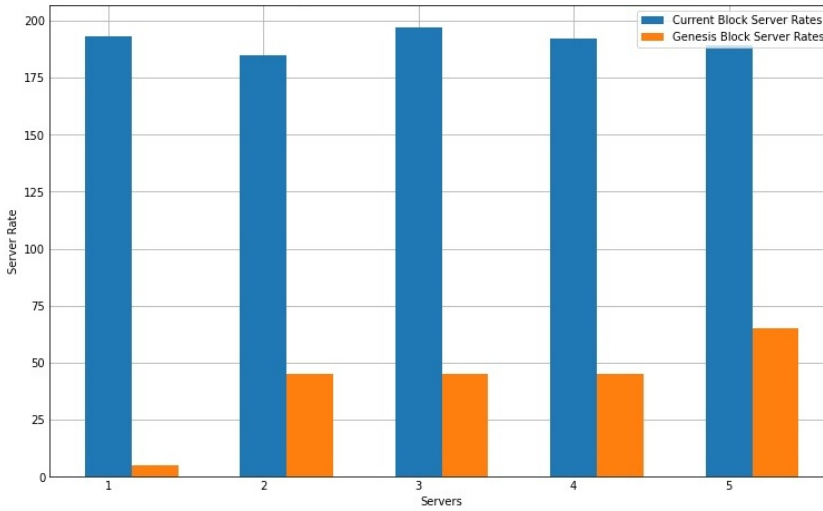

Figure 4. Comparison of current block vs genesis block

major factors that PSOR2B-BC hinge on for improved server rates are the consideration of request size and server/resource capacity; a resource/server whose capacity is low should not be saddled with a request whose size surpasses its capacity, likewise a server/resource with high capacity should not be lumbered with requests of minuscule sizes.

Figure 5 illustrates the PSOR2B-BC'S average transfer rate/throughput of three clusters with or without the incorporation of Hyperledger fabric BC. In other words, the figure shows the effects of the Hyperledger fabric BC has on the PSOR2B of [19]. From the illustration, it can be deduced that BC further improves transfer rate between nodes and layers, consequently improving resource optimisation thus bringing about overall system efficiency. This therefore implies that the QoS, and QoE are further improved.

The results depicted in Figures 6 and 7 illustrate the length of the chain against the transfer rate of five (5) servers/resources. From the pictorial representation of Figure 6, it can be deduced that as long as there is transfer of data between nodes, resultant of a request made, irrespective of the transfer rate, the length of the chain increases. A server with greater capacity will inevitably generate more data and consequently have a lengthier chain. It suffices therefore to say that the length of the chain is not determinant on the transfer rate, but rather on the transfer of information between nodes as they interact. From the results of the cumulative server performance, as depicted in Figure 7, on the other hand, it can be argued that, on the average, an increase in transfer rate, equates an increase in the chain length. Irrespective of ones stance, in the proposed PSOR2B-BC, the length of the chain is kept minimal by association of the Hyperledger fabric BC at the Dew layer

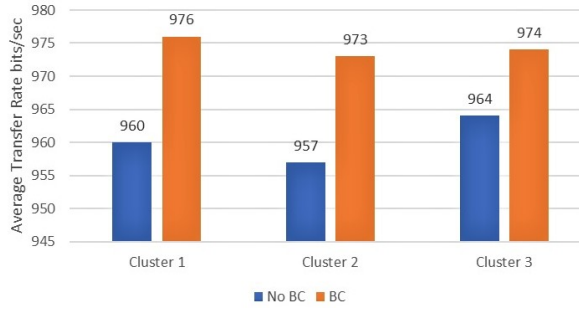

Figure 5. Average transfer rate

to control the records stored, propagated, and offloaded by the CH.

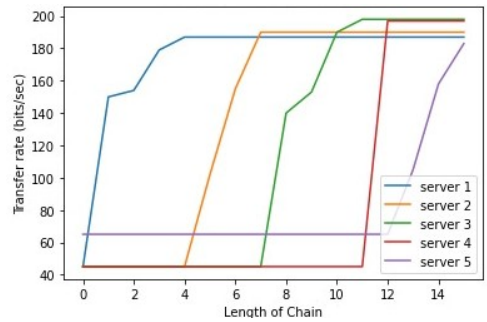

Figure 6. Individual server performance

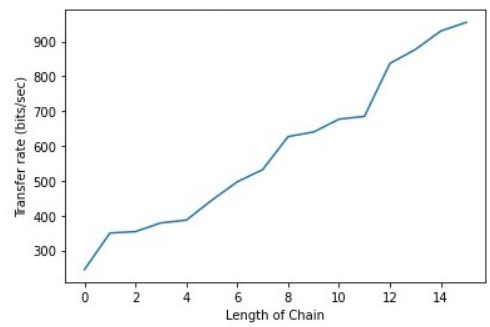

Figure 7. Cumulative server performance

The optimal performance consequent of frequent recordings of optimal paths for resource allocations is resultant of each iteration on the BC. By recording every new optimal path discovered on the network, the performance, as indicated by the blocks,

helps determine a server's capacity. Subsequent requests are routed based on the server's capacity thereby bringing about a secure and optimized resource allocation scheme. The hardware has no chain of course but the number of requests allocated to each server/resource are relevant, thus processed adequately, eliminating under or over utilization problems.

Existing systems seldom use the information stored in the CH of the PSOR2B-BC for appropriate scheduling and allocating of resources, this therefore affects the server rates, as the requests are randomly handled by servers/resources. This random approach can lead to over or under utilization of resources, thus impeding optimality. The work of [20] which proposed a decentralised Blockchain-Based Consumer Electronics for Data Sharing and Secure Payment platform that generates bills and provides incentive for legitimate consumers was compared to the proposed PSOR2B-BC owing to the similarities; security and data sharing. Comparison of [20]'s model, labelled BC-EDSSP to the PSOR2B-BC, as illustrated in Figure 8, indicates firstly the Hyperledger fabric's capacity to provide better optimization features for the IoT environment over the ethereum BC, and secondly, PSOR2B-BC's superiority over the BC-EDSSP model.

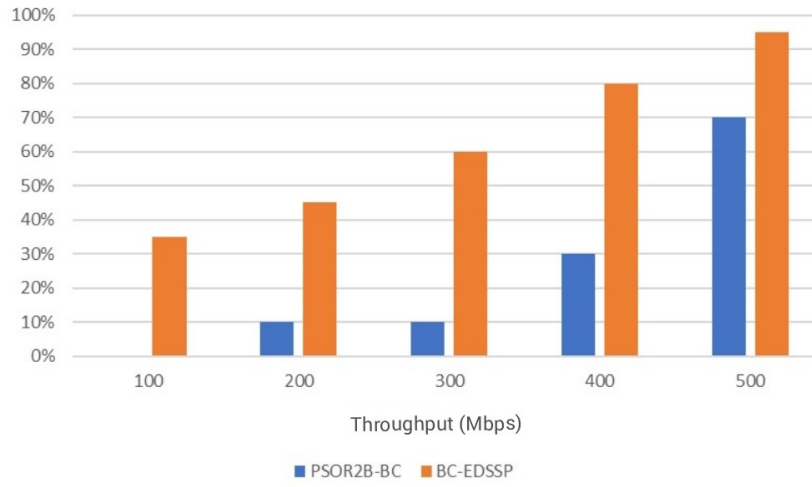

Figure 8. Public vs federated Blockchain

5 CONCLUSION

There are a plethora of challenges in making the IoT efficient and threat-free. On a technical side, IoT raises concerns related to its dynamic nature and context; security requirements and controls that are applicable to one IoT scenario do not necessarily work for different scenarios, or even the same scenario but different con-

text, as data enters or leaves, or when contexts are changing, security requirements may change. Recent years have seen an increase in the number of academic and industrial research works focusing on different research challenges related to this problem; there have been contributions to these lines of research through varying methodologies, yet, the current state of the art fails to provide evidence of appropriate approaches that supports dynamic security requirements analysis and reasoning. This therefore results in inflexible infrastructures, lost investments, damages resulting from mechanisms not matching the threats.

The explosive growth of IoT equates to ample data movement, and with that comes crucial requirements for resource optimization and security. This paper developed an efficient and secure resource optimization algorithm tagged PSOR2B-BC. The PSOR2B-BC utilizing a blend of adaptive algorithms, EC and BC paradigms to ensure the best of multiple worlds is harnessed, brings about a secure and optimal resource optimization mechanism for IoT systems. The proposed PSOR2B-BC's flexibility allows for applicability in all aspects of IoT resource optimization. In building the PSOR2B-BC, incorporation of selected attributes from the existing concepts paved way to avoiding most of the cons associated with exiting the algorithms, thus the PSOR2B-BC is more robust and efficient. Furthermore, the results generated by the proposed PSOR2B-BC show how the incorporation of BC further improves efficiency aside the prospective privacy and security measures it provides. The BC improves server rates by keeping records of optimal paths for routing of requests to appropriate resources, thus further improving the QoS and consequently the users' QoE.

Future works can further test the proposed PSOR2B-BC's security feature so as to evaluate the proposed policies and system design through various forms of security testing. Testing the security feature of the proposed model would imply logical assessment of BC's assorted operational elements. Additionally, owing to Hyperledger fabric BC's framework being quite new, there is great lack of proven use cases; therefore, future work can explore the deployment of the PSOR2B-BC model in real-life scenarios to further improve the trending societal opulence – IoT.

REFERENCES

[1] Kamran, M.—Khan, H. U.—Nisar, W.—Farooq, M.—Rehman, S. U.: Blockchain and Internet of Things: A Bibliometric Study. Computers & Electrical Engineering, Vol. 81, 2020, Art. No. 106525, doi: 10.1016/j.compeleceng.2019.106525.

[2] Lai, C. S.—Jia, Y.—Dong, Z.—Wang, D.—Tao, Y.—Lai, Q. H.—Wong, R. T. K.—Zobaa, A. F.—Wu, R.—Lai, L. L.: A Review of Technical Standards for Smart Cities. Clean Technologies, Vol. 2, 2020, No. 3, pp. 290–310, doi: 10.3390/cleantechnol2030019.

[3] Banafa, A.: A Secure Model of IoT with Blockchain. BBVA OpenMind, 2016, https://www.bbvaopenmind.com/en/technology/digital-world/a-secure-model-of-iot-with-blockchain/.

[4] Li, X.—Xu, L. D.: A Review of Internet of Things – Resource Allocation. IEEE Internet of Things Journal, Vol. 8, 2021, No. 11, pp. 8657–8666, doi: 10.1109/JIOT.2020.3035542.

[5] Xu, L. D.—Lu, Y.—Li, L.: Embedding Blockchain Technology into IoT for Security: A Survey. IEEE Internet of Things Journal, Vol. 8, 2021, No. 13, pp. 10452–10473, doi: 10.1109/JIOT.2021.3060508.

[6] Ali, M. S.—Vecchio, M.—Pincheira, M.—Dolui, K.—Antonelli, F.—Rehmani, M. H.: Applications of Blockchains in the Internet of Things: A Comprehensive Survey. IEEE Communications Surveys & Tutorials, Vol. 21, 2019, No. 2, pp. 1676–1717, doi: 10.1109/COMST.2018.2886932.

[7] Zhao, H.—Deng, S.—Zhang, C.—Du, W.—He, Q.—Yin, J.: A Mobility-Aware Cross-Edge Computation Offloading Framework for Partitionable Applications. 2019 IEEE International Conference on Web Services (ICWS), 2019, pp. 193–200, doi: 10.1109/ICWS.2019.00041.

[8] Rehan, M. M.—Rehmani, M. H.: Blockchain-Enabled Fog and Edge Computing: Concepts, Architectures, and Applications. CRC Press, 2021.

[9] Iansiti, M.—Lakhani, Karim, R.: The Truth About Blockchain. Harvard Business Review, 2017, `https://hbr.org/2017/01/the-truth-about-blockchain`.

[10] Dorri, A.—Steger, M.—Kanhere, S. S.—Jurdak, R.: BlockChain: A Distributed Solution to Automotive Security and Privacy. IEEE Communications Magazine, Vol. 55, 2017, No. 12, pp. 119–125, doi: 10.1109/MCOM.2017.1700879.

[11] Yu, Y.—Li, Y.—Tian, J.—Liu, J.: Blockchain-Based Solutions to Security and Privacy Issues in the Internet of Things. IEEE Wireless Communications, Vol. 25, 2018, No. 6, pp. 12–18, doi: 10.1109/MWC.2017.1800116.

[12] Kshetri, N.: Can Blockchain Strengthen the Internet of Things? IT Professional, Vol. 19, 2017, No. 4, pp. 68–72, doi: 10.1109/MITP.2017.3051335.

[13] Dorri, A.—Kanhere, S. S.—Jurdak, R.—Gauravaram, P.: Blockchain for IoT Security and Privacy: The Case Study of a Smart Home. 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), 2017, pp. 618–623, doi: 10.1109/PERCOMW.2017.7917634.

[14] Huckle, S.—Bhattacharya, R.—White, M.—Beloff, N.: Internet of Things, Blockchain and Shared Economy Applications. Procedia Computer Science, Vol. 98, 2016, pp. 461–466, doi: 10.1016/j.procs.2016.09.074.

[15] Cachin, C.: Architecture of the Hyperledger Blockchain Fabric. Workshop on Distributed Cryptocurrencies and Consensus Ledgers (DCCL 2016), 2016, pp. 1–4, `https://www.zurich.ibm.com/dccl/papers/cachin_dccl.pdf`.

[16] Puttaswamy, K. P. N.—Kruegel, C.—Zhao, B. Y.: Silverline: Toward Data Confidentiality in Storage-Intensive Cloud Applications. Proceedings of the 2nd ACM Symposium on Cloud Computing (SOCC '11), 2011, doi: 10.1145/2038916.2038926.

[17] Akwirry, B.—Bessis, N.—Malik, H.—McHale, S.: A Multi-Tier Trust-Based Security Mechanism for Vehicular Ad-Hoc Network Communications. Sensors, Vol. 22, 2022, No. 21, Art. No. 8285, doi: 10.3390/s22218285.

[18] Wang, W.—Hoang, D. T.—Hu, P.—Xiong, Z.—Niyato, D.—Wang, P.—Wen, Y.—Kim, D. I.: A Survey on Consensus Mechanisms and Mining Strategy

Management in Blockchain Networks. IEEE Access, Vol. 7, 2019, pp. 22328–22370, doi: 10.1109/ACCESS.2019.2896108.

[19] Datiri, D. D.—Li, M.: Effects of Particle Swarm Optimisation on a Hybrid Load Balancing Approach for Resource Optimisation in Internet of Things. Sensors, Vol. 23, 2023, No. 4, Art. No. 2329, doi: 10.3390/s23042329.

[20] William, A. D. J.—Rajendran, S.—Pranam, P.—Berry, Y.—Sreedharan, A.—Gul, J.—Anand, P.: Blockchain Technologies: Smart Contracts for Consumer Electronics Data Sharing and Secure Payment. Electronics, Vol. 12, 2023, No. 1, Art. No. 208, doi: 10.3390/electronics12010208.

[21] Datiri, D. D.—Li, M.: Load Balancing for Resource Optimization in Internet of Things (IoT) Systems. Computing and Informatics, Vol. 41, 2022, No. 6, pp. 1425–1445, doi: 10.31577/cai_2022_6_1425.

[22] Danzi, P.—Kalør, A. E.—Stefanović, C.—Popovski, P.: Delay and Communication Tradeoffs for Blockchain Systems with Lightweight IoT Clients. IEEE Internet of Things Journal, Vol. 6, 2019, No. 2, pp. 2354–2365, doi: 10.1109/JIOT.2019.2906615.

[23] Yu, S.—Lv, K.—Shao, Z.—Guo, Y.—Zou, J.—Zhang, B.: A High Performance Blockchain Platform for Intelligent Devices. 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN), 2018, pp. 260–261, doi: 10.1109/HOTICN.2018.8606017.

[24] Orman, H.: Blockchain: The Emperors New PKI? IEEE Internet Computing, Vol. 22, 2018, No. 2, pp. 23–28, doi: 10.1109/MIC.2018.022021659.

[25] Wang, R.—He, J.—Liu, C.—Li, Q.—Tsai, W. T.—Deng, E.: A Privacy-Aware PKI System Based on Permissioned Blockchains. 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), 2018, pp. 928–931, doi: 10.1109/ICSESS.2018.8663738.

[26] Sharma, V.: An Energy-Efficient Transaction Model for the Blockchain-Enabled Internet of Vehicles (IoV). IEEE Communications Letters, Vol. 23, 2019, No. 2, pp. 246–249, doi: 10.1109/LCOMM.2018.2883629.

[27] Strawn, G.: BLOCKCHAIN. IT Professional, Vol. 21, 2019, No. 1, pp. 91–92, doi: 10.1109/MITP.2018.2879244.

[28] Henry, R.—Herzberg, A.—Kate, A.: Blockchain Access Privacy: Challenges and Directions. IEEE Security & Privacy, Vol. 16, 2018, No. 4, pp. 38–45, doi: 10.1109/MSP.2018.3111245.

[29] Alrehaili, A.—Namoun, A.—Tufail, A.: A Comparative Analysis of Scalability Issues Within Blockchain-Based Solutions in the Internet of Things. International Journal of Advanced Computer Science and Applications, Vol. 12, 2021, No. 9, pp. 480–490, doi: 10.14569/IJACSA.2021.0120955.

[30] Xiao, K.—Geng, Z.—He, Y.—Xu, G.—Wang, C.—Cheng, W.: A Blockchain Based Privacy-Preserving Cloud Service Level Agreement Auditing Scheme. In: Yu, D., Dressler, F., Yu, J. (Eds.): Wireless Algorithms, Systems, and Applications (WASA 2020). Springer, Cham, Lecture Notes in Computer Science, Vol. 12384, 2020, pp. 542–554, doi: 10.1007/978-3-030-59016-1_45.

[31] Deng, S.—Xiang, Z.—Zhao, P.—Taheri, J.—Gao, H.—Yin, J.—

Zomaya, A. Y.: Dynamical Resource Allocation in Edge for Trustable Internet-of-Things Systems: A Reinforcement Learning Method. IEEE Transactions on Industrial Informatics, Vol. 16, 2020, No. 9, pp. 6103–6113, doi: 10.1109/TII.2020.2974875.

[32] Yli-Huumo, J.—Ko, D.—Choi, S.—Park, S.—Smolander, K.: Where Is Current Research on Blockchain Technology? – A Systematic Review. PLoS ONE, Vol. 11, 2016, No. 10, Art. No. e0163477, doi: 10.1371/journal.pone.0163477.

[33] Fitzi, M.—Gaži, P.—Kiayias, A.—Russell, A.: Parallel Chains: Improving Throughput and Latency of Blockchain Protocols via Parallel Composition. 2018, https://eprint.iacr.org/2018/1119.

[34] Ouaddah, A.—Abou Elkalam, A.—Ait Ouahman, A.: FairAccess: A New Blockchain-Based Access Control Framework for the Internet of Things. Security and Communication Networks, Vol. 9, 2016, No. 18, pp. 5943–5964, doi: 10.1002/sec.1748.

[35] Tariq, U.—Ibrahim, A.—Ahmad, T.—Bouteraa, Y.—Elmogy, A.: Blockchain in Internet-of-Things: A Necessity Framework for Security, Reliability, Transparency, Immutability and Liability. IET Communications, Vol. 13, 2019, No. 19, pp. 3187–3192, doi: 10.1049/iet-com.2019.0194.

[36] Zhai, X.—Pang, S.—Wang, M.—Qiao, S.—Lv, Z.: TVS: A Trusted Verification Scheme for Office Documents Based on Blockchain. Complex & Intelligent Systems, Vol. 9, 2023, No. 3, pp. 2865–2877, doi: 10.1007/s40747-021-00617-1.

[37] Pang, S.—Li, X.—Xiong, N. N.—Wang, M.—Qiao, S.: Optimal Target User Selection Policy for D2D Wireless Caching Networks. IEEE Transactions on Network Science and Engineering, Vol. 8, 2021, No. 3, pp. 2665–2678, doi: 10.1109/TNSE.2021.3103896.

[38] Jia, X.—Lei, T.—Du, X.—Liu, S.—Meng, H.—Nandi, A. K.: Robust Self-Sparse Fuzzy Clustering for Image Segmentation. IEEE Access, Vol. 8, 2020, pp. 146182–146195, doi: 10.1109/ACCESS.2020.3015270.

[39] Pang, S.—Wang, N.—Wang, M.—Qiao, S.—Zhai, X.—Xiong, N. N.: A Smart Network Resource Management System for High Mobility Edge Computing in 5G Internet of Vehicles. IEEE Transactions on Network Science and Engineering, Vol. 8, 2021, No. 4, pp. 3179–3191, doi: 10.1109/TNSE.2021.3106955.

[40] Yu, Y.—Liu, S.—Yeoh, P. L.—Vucetic, B.—Li, Y.: LayerChain: A Hierarchical Edge-Cloud Blockchain for Large-Scale Low-Delay Industrial Internet of Things Applications. IEEE Transactions on Industrial Informatics, Vol. 17, 2021, No. 7, pp. 5077–5086, doi: 10.1109/TII.2020.3016025.

Dorcas Dachollom Datiri received her Ph.D. from the Department of Electronic and Electrical Engineering, Brunel University, London, UK in 2024. She has been a Graduate Teaching Assistant at the same institution (2019-2024) and a Lecturer at the University of Jos, Nigeria (2015-2019). Her research interests are data science, data analytics, optimization techniques, edge and cloud computing, blockchain technologies, cybersecurity, Internet of Things and Industry 4.0. She is an Associate Fellow of Higher Education, UK, and member of the Internet Society Community (ISOC) and NACOSS professional bodies.

Maozhen Li is Professor in the Department of Electronic and Electrical Engineering, Brunel University, London, UK. He received his Ph.D. from the Institute of Software, Chinese Academy of Sciences in 1997. He was Post-Doctoral Research Associate in the School of Computer Science and Informatics at Cardiff University, UK in 1999–2002. His main research interests include high performance computing, big data analytics and intelligent systems with applications to smart grid, smart manufacturing and smart cities. He has over 200 research publications in these areas including 4 books. He has served over 30 IEEE conferences and is on the editorial board of a number of journals. He is Fellow of the British Computer Society (BCS) and the Institute of Engineering and Technology (IET).