A Novel Blockchain-as-a-Service (BaaS) Platform for Local 5G Operators

Nisita Weerasinghe, *Student Member, IEEE,* Tharaka Hewa, *Student Member, IEEE,* Madhusanka Liyanage, *Senior Member, IEEE,* Salil S. Kanhere, *Senior Member, IEEE,* and Mika Ylianttila, *Senior Member, IEEE*

***Abstract*—5G is a promising technology that has the potential to support verticals and applications such as Industrial Internet of Things IoT (IIoT), smart cities, autonomous vehicles, remote surgeries, virtual and augmented realities, and so on. These verticals have a diverse set of network connectivity requirements, and it is challenging to deliver customized services for each by using a common 5G infrastructure. Thus, the operation of Local 5G operator (L5GO) networks or private 5G networks are a viable option to tackle this challenge. A L5GO network is a localized small cell network which can offer tailored service delivery. The adaptation of network softwarization in 5G allows vertical owners to deploy and operate L5GO networks. However, the deployment of L5GOs raises various issues related to management of subscribers, roaming users, spectrum, security,and also the infrastructure. This paper proposes a blockchain-based platform to address these issues. The paper introduces a set of blockchain-based modularized functions such as service rating systems, bidding techniques, and selection functions, which can be used to deploy different services for L5GOs. Exploitation of blockchain technology ensures availability, non-reliance on trusted third parties, secure transfer payments, and stands to gain many more advantages. The performance and the viability of the proposed platform are analyzed by using simulations and a prototype implementation.**

***Index Terms*—5G, Local 5G operators, Blockchain, Smart Contracts**

TABLE I: Summary of Important Acronyms.

Acronym	Definition
5G	Fifth Generation
APF	Agreement establishment and Payment settlement Function
BaaS	Blockchain as Service
CDR	Call Detail Records
DApps	Decentralized Applications
DCH	Data Clearing House
DLT	Distributed Ledger Technology
DMF	Data Management Function
FPF	Fraud Prevention Function
IMSI	International Mobile Subscriber Identity
HPMN	Home Public Mobile Network
IoT	Internet of Things
L5GO	Local 5G operator
MEC	Multi-access Edge Computing
MF	Marketplace Function
MNO	Mobile Network Operator
MQTT	Message Queuing Telemetry Transport
NF	Network Functions
NFV	Network Function Virtualization
PoW	Proof of Work
QoS	Quality of Service
RMF	Reputation Management Function
RPC	Remote Procedure Call
SF	Selection Function
TS	Traditional System
SMF	Subscription Management Function
VM	Virtual Machine
VNF	Virtual Network Function
VPMN	Visited Public Mobile Network

I. Introduction

Within the application context of the modern telecommunication ecosystem, high data consumption is a vital requirement. In the future, the number of smart devices connected to one person will increase with the beginning of the 5G era [1]. As a result, the network capacity requirement grows significantly, and the network operators must deliver the network services to end users with minimal latency, ultra-high speed, and ultra-reliability. Network operators try to build up the network systems which could serve all of those hungry endpoints. This can be considered one of the major challenges that a network operator has to face. As a solution to fulfill this requirement, 5G researchers are looking for new frequencies. For instance, 5G is trying to operate in high frequency that is in mm wave lengths. However, signal absorption is high for mmWave frequencies and the operating range goes low. Therefore, research has proposed to establish small cells close to each other while maintaining isolation between them [2], [3].

Ultra-dense deployment of 5G base stations, especially indoors by traditional or incumbent Mobile Network Operators (MNOs) would be significantly challenging due to the relative costs involved in such deployments, especially considering the multitude of incumbents currently available. This has led to the development of a new type of network management and service provisioning paradigm called Local 5G operator (L5GO) networks, or private 5G networks. L5GO allows companies and local governments to operate their own 5G communication ecosystems with a unique design depending upon the operation-specific requirements [4], [5]. L5GOs can be used to accelerate the digital innovation in various

Nisita Weerasinghe, Tharaka Hewa and Mika Ylianttila are with the Center for Wireless Communications, University of Oulu, Finland. e-mail: {firstname.lastname}@oulu.fi

Madhusanka Liyanage is with the School of Computer Science, University College Dublin, Ireland and the Center for Wireless Communications, University of Oulu, Finland. e-mail:madhusanka@ucd.ie

Salil S. Kanhere is with School of Computer Science and Engineering, University of New South Wales, Sydney, Australia. Email: salil.kanhere@unsw.edu.au

fields such as hospitals, factories, industries, universities, and shopping malls. Further, the contrasting features of L5GOs compared with MNO are exhibited in Fig. 1.

However, the deployment of L5GOs raises various challenges related to roaming users, spectrum, security, management of subscribers, and infrastructure. These issues need to be addressed in order to obtain the maximum benefits of L5GO deployments. The critical challenges encountered with the present systems include lack of transparency in roaming and resource-sharing procedures, violation of pre-agreements by network operators, failure to offer high quality service as expected, and abuse of user identity information. Another major challenge is use of static agreements to accommodate extensive numbers of subscribers real time in a 5G domain, which causes delay in processing agreements. Also, monitoring agreement violations and imposing dynamic penalty schemes are challenging in the current systems.

Blockchain technology converts the traditional way of our work by allowing users to exclude the central authority from various services, cutting costs and uplifting productivity [6]. The cost cutting is applied when blockchain operates in a private mode. Blockchain can also be comprehended as a decentralized ledger. The technology adds transactions to the ledger after being validated by miners in the blockchain network rather than by a single authorization unit [7]. Thus, the immutability within the blockchain records and blocks, and none of a party could forge the data easily [8], [9]. Moreover, blockchain-based smart contracts can enable distributed and trusted automated services [10], [11]. Due to these properties, blockchain and smart contacts are utilized in many telecommunication applications—for instance, in addressing security and privacy issues in different 5G services [12], assurance of trust between mobile operators, and enabling transparency in pre-defined agreements [13], replacement of roaming agreements with smart contracts and elimination of dependent on intermediary parties in the transactions [14], and introducing blockchain-based solutions to mitigate roaming fraud [15], [16]. Thus, blockchain and smart contracts can be a viable solution to resolve the existing implementation and management challenges in L5GO networks.

To mitigate challenges encountered in L5GO ecosystems, this paper proposes a novel Blockchain-as-a-Service (BaaS) platform. The distinct features of our work include significant value-added services in the L5GO context. For instance, we propose the implementation of a service-quality evaluation scheme by maintaining a smart contract operated rating system, along with an incentive-penalty scheme. In addition to that, we propose the establishment of a dynamic agreement system to cater the user requirements, in real-time. Furthermore, we propose the deployment of selection algorithms to discover the optimal service provider to each customer and to enhance their quality of experience. Moreover, assurance of trust and privacy with blockchain is one of our key focus points in managing subscription details, to avoid subscription theft and use of subscriber details unlawfully. We also suggest the facilitation of secure payment transactions between providers and users to eradicate fraudulent practices. Another distinguishable feature of our work is the implementation of roaming fraud prevention techniques to minimize the occurrence of fraud during roaming instances. Also, our system guarantees the security of IoT data with the enforcement of decentralized access control through smart contracts. Finally, the proposed architecture addresses the issues related to capacity heterogeneity in IoT nodes by accommodating storage facilities in the distributed ledger.

The contributions of our study can be summarized as follows:

- Proposes Blockchain-as-a-Service (BaaS) platform to address the key challenges within a L5GO ecosystem
- Proposes novel blockchain-based modularized functions to enable L5GO related services efficiently
- Evaluates the proposed architecture in a simulated environment and verify the feasibility via a prototype implementation

The rest of the paper is organized as follows: Section II highlights the current challenges in L5GOs, while Section III examines existing works. Section IV proposes the novel architecture, and Section V discusses its key functions. Section VI presents enabled services using the introduced approach. Section VII elaborates on the developed simulation setup and test results. Section VIII presents the prototypical implementation. Section IX provides the experimental results. Finally, Section X concludes the paper. Table I includes a summary of important acronyms.

II. Existing Challenges in L5GOs

This section presents the main challenges in the L5GO ecosystem which can be resolved by using blockchain and smart contracts.

A. Spectrum Sharing

By default, the mobile network spectrum is restricted and the demand is expected to inflate with the expansion of future computing and networking demands. Therefore, the spectrum management techniques are expected to advance by virtue of the administrative allocation approach to market-based technique and the unlicensed commons technique. Administrative allocation refers to when a regulatory authority determines the party that is eligible to utilize the spectrum. However, according to the market-based mechanism, the regulator is responsible to specify spectrum property rights offered by market methods (e.g., Auction), whereas in the commons approach spectrum sharing is permitted under the policies defined by the regulator. Market development dominates traditional spectrum management mechanisms since most of the vertical markets are willing to deploy L5GOs deprived of direct MNO connections. In the L5GO concept, there are three spectrum management options for a L5GO listed in the research study [17]. These are MNO-centric, collaboration-centric and local operator–centric techniques. The MNO-centric technique refers to when MNOs deploy L5GOs in their prevailing licensed spectrum bands. Another spectrum assignment model is sharing existing MNO bands with L5GOs to deploy 5G networks that can satisfy the needs of vertical markets; this is known as the Collaboration-centric model.

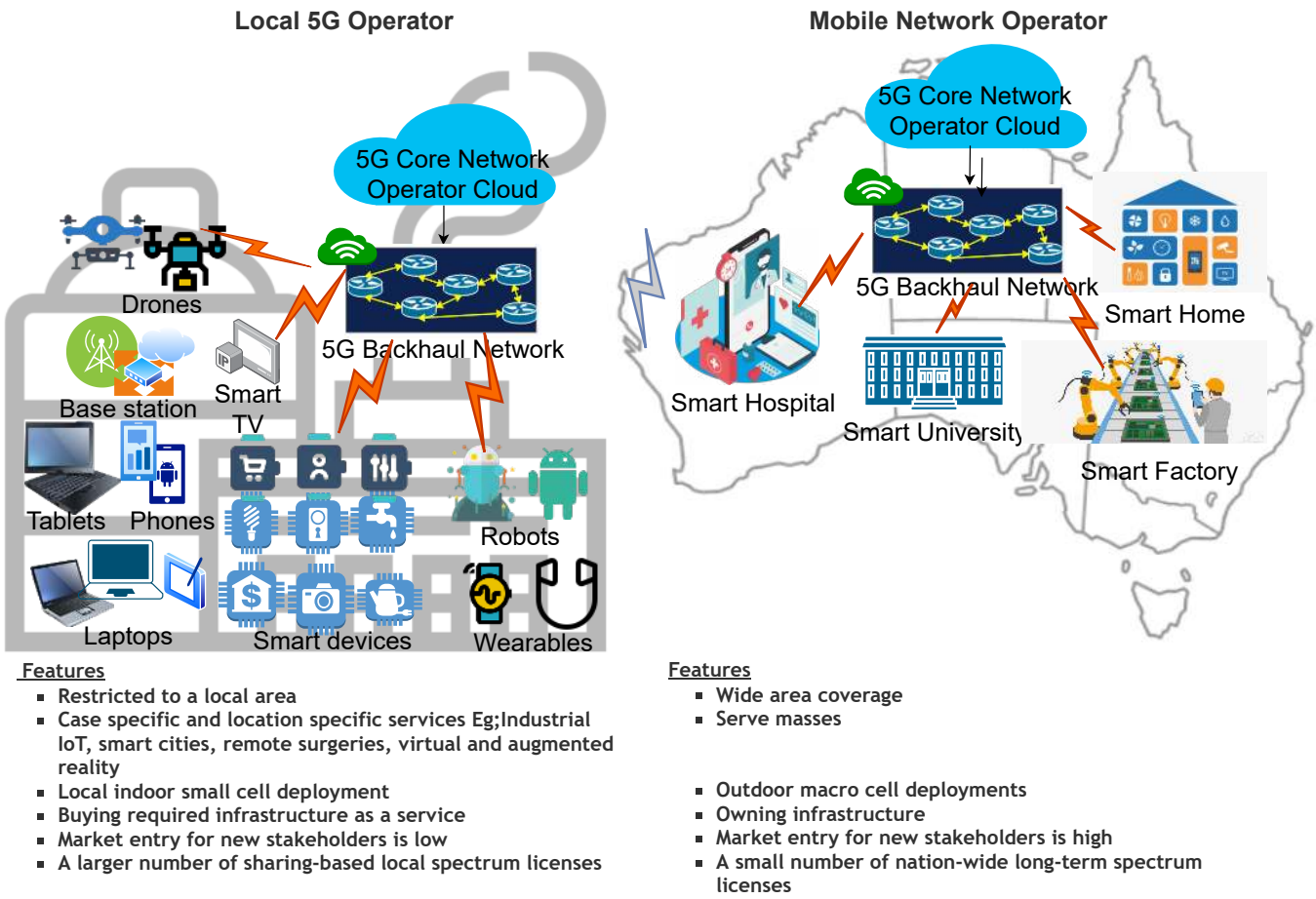


Fig. 1: Comparison between MNO and L5GO

Introduction of local spectrum licensing to establish local 5G networks to cater to the specific requirements requested by vertical sectors is called the local operator–centric approach.

The latest trend in spectrum management has become the assignment of local spectrum licenses: the growth of 5G networks has recently evolved from the legacy MNO-centric model to the local operator model. Distinctive challenges were foreseen with the deployment of L5GO models. Both these models incorporate with two stakeholders. That is, the synergy of MNO and L5GO builds the Collaboration-centric model, whereas the local operator model consists of regulator and L5GO parties. Therefore, a centralized authority is functioning to handle all the collaboration-related operations and the agreements. This setup adds an overhead to both of the parties and the service subscribers incurred with extra fees for the intermediary party.

B. Roaming

Roaming in L5GO connects the home network operator with another network domain when the operator does not have proper coverage within the geographical region. Currently, home MNO or L5GO have pre-established agreements with visitor MNOs enforcing the negotiations and policies to activate the roaming services for its customers.

The accepted link from a specific partner operator might deliver modest coverage and alterations in the package prices time to time, causing the user experience to be negatively impacted. Further, the violation of pre-agreements by network operators leads to lack of transparency in the roaming processes and causes bill-shocks [18] to users. Moreover, roaming fraud alone costs the telecommunication industry over USD 38 billion every year [19]. For an instance, over-utilization, one of the most commonly executed frauds, exploits the delay of transferring Call Detail Records (CDR) information to the Home Public Mobile Network (HPMN) by the Visited Public Mobile Network (VPMN) when the subscriber is roaming. While the majority of fraud schemes are still prevalent, industry has been struggling to remedy those with orthodox techniques available today.

C. Offloading

Offloading allows MNOs or L5GOs to hand over the network traffic load to other networks, boosting the network efficiency of the system, minimizing the power consumption of base stations, achieving expected QoS (Quality of Service),

maximizing throughput, providing high bandwidth, and many more benefits. Since L5GOs offer better coverage inside their premises, MNOs can use these L5GOs to serve their subscribers when they reside in a L5GO's coverage area.

With the popularity of L5GOs, there will be more customers attracted to its service. The smart city is a potential application for L5GO. A massive number of tenants expected to onboard with an extensive usage traffic. This phenomenon causes low network efficiency in the system and maximizes the power consumption of base stations [20]. This will degrade the service quality and throughput of the system. Therefore, offloading is an ideal technique to eradicate the significant drawbacks in terms of scaling up the usage. However, there are potential challenges that must be addressed in the selection process of an L5GO to offload. This is because in the current system there is no real-time rating system to evaluate the performance of L5GOs. Also, manual selection of an L5GO will be challenging as they increase. Therefore, there is a high-demand requirement for dynamic selection of the best L5GO.

D. Infrastructure Sharing

Generally, L5GO contributes to the massive scaling requirements of subscribers and supports MNOs with customized demand varieties of their customers by providing cost-effective local service. To strengthen the service, L5GO are required to collaborate with small-scale or third-party providers such as content providers, network infrastructure vendors, equipment vendors, and facility owners [21].

For an efficient collaboration, the existence of a middle organization is essential to handle the agreements and consequences where both the L5GO and third party providers must pay additional fees. This causes additional overheads, especially for smaller business entities. There will be extra processing and transaction since all the agreements need to go through an intermediary party.

E. Subscription Management

Subscription management includes managing the stack of value-added services based on each subscriber's subscription criteria. Significant current challenges in subscription management include identity or subscription ID theft. A malicious node deliberately uses a legitimate user's identity credentials to consume data or access to their respective registered L5GO. In addition to that, the subscribers are required to infiltrate a sequence of authenticating checkpoints whenever they visit another L5GO, which is a cumbersome experience for the customer. Furthermore, subscription information sharing is limited within other operators in the classical network ecosystems.

F. Virtual Network Function (VNF) Management

The collaboration of NFV (Network Function Virtualization) and MEC (Multi-access Edge Computing) contributes to achieving 5G networking by moving VNF to the edge. This process of migration and complete management procedures is vulnerable to security challenges.

Generally, several organizations operate the NFV ecosystem. Consequently, challenges might be triggered if any illegal organization used VNF instances. This incurs massive damage to VNF and generic hardware provider. Furthermore, more problems arise when the services delivered by different VNF vendors are not compatible as promised. For instance, false details on a VNF's consumption and payment policy disputes. Additionally, no there is no prevailing method of measuring the reputation of each VNF provider before getting acquiring their services. There are also challenges in the payment settlement process between VNF provider and the L5GO [34].

G. Internet of Things (IoT) data Management

IoT has become an integral part of the current generation of information technology and it continues to grow at a rapid pace. As data generation, data analysis, and data transportation are at the heart of IoT, it is equally important to secure them throughout their life-cycle.

Due to the centralized nature of the majority of IoT systems available today, they will not be able to accommodate the exponential growth of IoT technology expected in the near future [35]. Data security will be at a risk and devices will have to suffer from increased latencies due to network bottlenecks.

III. Related Work

Up to now, various approaches have been evolving to investigate how blockchain can be utilized to facilitate 5G services. Among them, we focus first on the research studies related to blockchain-based, spectrum-sharing applications. In [23], the practicality of employing the smart contract assisted sharing was evaluated based on decentralization, transparency, immutability, availability, and security. [22] proposed a blockchain-based spectrum sharing scheme combined with game theory applications to develop the ideal sharing strategy. Then, the authors proposed boosting the spectrum sharing utilization rate of operators and to cut off the extra costs paid for the trading party. In addition, the consortium chain architecture was utilized for user authentication and to track transaction details, which ensure that no party could manipulate the recorded data. Multi-operator spectrum sharing was enabled in [24], with the use of a permissioned blockchain, adopting a PBFT consensus algorithm to leverage the high throughput and to reduce the high block verification delay.

With regard to the roaming and offloading facilities, in [14], a smart contract is written to settle and notify the roaming charges between HPMN and VPMN. Moreover, a blockchain-based user balance transfer through online and offline means is proposed. Another literature study [25] proposed a blockchain-based architecture for a roaming platform and carried out a case study to analyze its performance from both the operator's and user's perspective. A blockchain-based roaming fraud prevention framework was proposed in [15]; this approach minimizes the data exchange delay and the excess cost with the replacement of DCH with the blockchain. Also, an economic model based on Stackelberg game was developed to maximize the benefits for users by allowing them to participate in the consensus process and earn extra profits for their involvement.

TABLE II: Comparison with Related Works

Features	[22]	[23]	[24]	[14]	[25]	[15]	[26]	[27]	[28]	[29]	[30]	[31]	[32]	[33]	Ours
Universal Wallet	No	No	Yes	Yes	Yes	No	No	Yes	Yes	No	No	No	No	No	Yes
Universal Identity	No	No	Yes	Yes	No	No	Yes	Yes	No	No	No	Yes	Yes	No	Yes
Auditable Auction	No	No	Yes	No	No	No	No	No	Yes	No	No	Yes	No	No	Yes
Roaming Fraud Prevention	No	No	No	No	No	Yes	No	No	No	No	No	No	No	No	Yes
Decentralized Traceability	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Load Balancing Technique	No	No	No	No	No	No	No	No	No	No	No	No	No	No	Yes
Service Quality Assessment	No	No	Yes	No	No	No	No	No	No	No	No	No	No	No	Yes

By evaluating previous studies on mobile subscriber management in 5G along with blockchain, [26] suggested a confidentiality enabling client identity management scheme involving blockchain technology. It was applied for both attribution and obscurity, and contributes to the entire process, from consumer registration to custom billing. The proposed system in [27] comprises four phases to provide reliable authentication and key agreement protocol for 5G networks: namely, initialization, registration, mining process, and authentication and key agreement protocol. In addition, this approach has the ability to tolerate most of the common attacks.

With regard to the prevalent research on VNF management, a blockchain-based reverse auction strategy was executed in [28] to promotes a rivalry between infrastructure suppliers to facilitate the VNF requirements of an end user. In [29], a blockchain-based platform was proposed to deliver tailored services to multi-tenants by chaining VNF between rival infrastructure providers, guaranteeing security in network slices.

In the same vein as other studies on blockchain-based infrastructure supply, [30] introduced a decentralized E-marketplace framework, combining blockchain technology to enhance the client experience via providing them cost-effective products based on their requirements. Moreover, general consequences caused with the use of public or private blockchain were dealt with in [31] by introducing an innovative framework that includes a hybrid of private and public blockchains. In this approach, private blockchain is permitted to handle vulnerable bids and given the sole permission for the auctioneer to discover the bids, while public blockchain was responsible for broadcasting the winner of the auction and to make payments liable.

Turning to IoT data management solutions, [32] proposed blockchain-based certificate issuance for IoT devices and retrieval of stored data via certificates, to achieve consumer confidentiality along with data reliability. A decentralized IoT data management scheme was implemented in [33] to mainly ensure the transparency of user data. Furthermore, their system facilitates storage of encrypted data in the blockchain while raw data is stored in a secure storage platform, to guarantee data privacy and integrity. The proposed model was able to overcome the issues generated with the centralized nature of the current IoT data management system.

Table II compares the proposed model with pertinent current solutions. This table proves the uniqueness of our methodology.

IV. Proposed Blockchain-as-a-Service (BaaS) Architecture

We propose a novel Blockchain-as-a-Service (BaaS) architecture for the L5GO ecosystem to overcome each of the potential challenges are explicitly described in Section II. This section explains the proposed BaaS architecture in detail.

The proposed BaaS architecture operates as an overlay entity which is spread across the L5GO ecosystem. An overlay blockchain will be utilized to provide blockchain-based services proposed in the BaaS architecture. This blockchain can be implemented in two different ways: as a public blockchain and as a consortium blockchain. In the ***public blockchain*** implementation, it is possible to utilize the existing blockchain platforms (e.g., Ethereum) to implement the services proposed in the BaaS architecture. However, this is expensive as the operation cost could increase with the value of the digital currency. Moreover, operational latency can also increase with the congestion of the network.

Therefore, we propose to use ***consortium blockchain*** for the BaaS architecture, as reflected in Fig. 2. Each stakeholder (i.e, MNOs, L5GOs, VNF providers, IoT tenants and cloud service providers) of the L5GO ecosystem can participate in maintaining the blockchain: they can deploy their own blockchain nodes (i.e., miners, full nodes, or light nodes), as illustrated in Fig. 2.

The blockchain deployment model can be customized as per the requirements of each stakeholder. For instance, the deployment setup of mining nodes and peer nodes can be defined as per the requirement. MNOs and L5GOs are operable as miner nodes which perform mining and peer transactions. The VNF and cloud service providers can be operated as miner nodes since they have enough resources. The corresponding blockchain nodes for IoT nodes can be deployed on fog computing nodes which may be comparably less in computing power. In such a case, the IoT tenants blockchain nodes are only operable as full nodes in the blockchain that perform transactions committing to the network.

Moreover, the blockchain node assigned to each stakeholder is capable of performing the customized services in the system. For instance, the blockchain node in IoT tenants can handle the IoT data management services to share with third-party services via the smart contracts. The key benefits of the integration of blockchain nodes to fulfill the services include the capability of handling comparably higher volumes of transactions in contrast with cloud-oriented architectures, and eliminating latency by the local blockchain node. The cloud

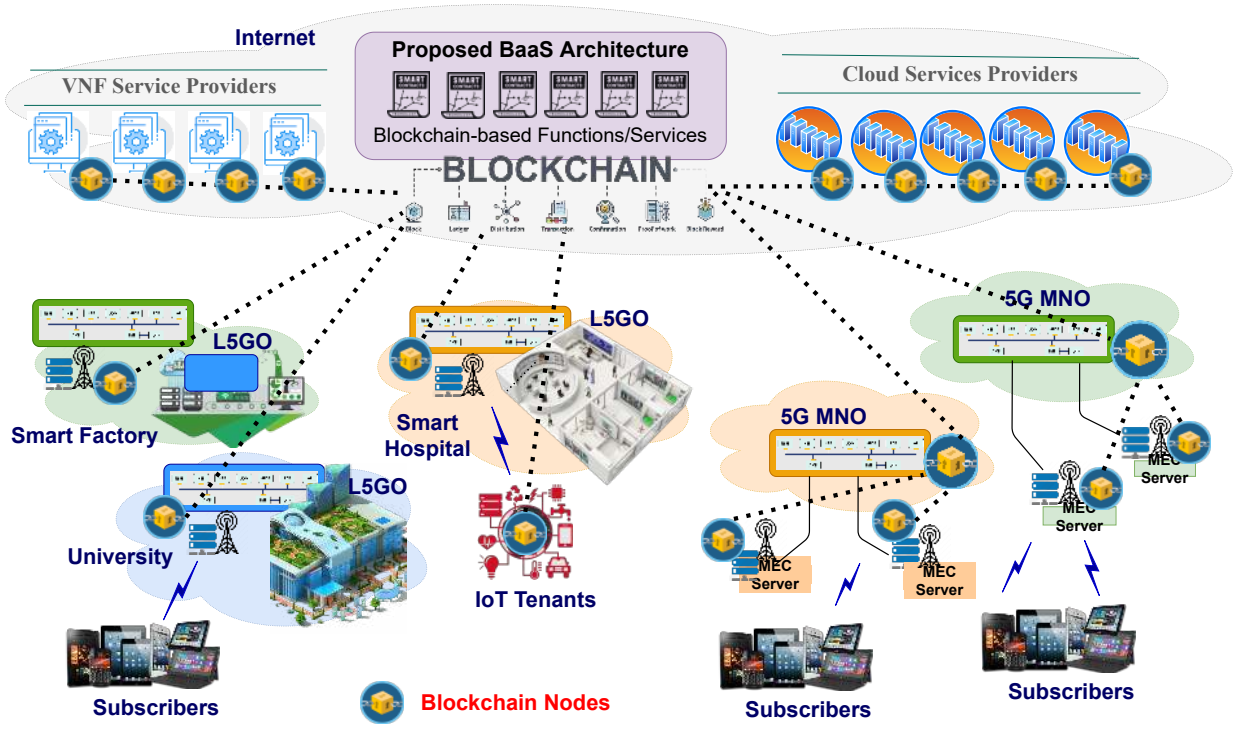


Fig. 2: The deployment of blockchain for the BaaS architecture

service invocation includes a data transit leg over the internet and forms a bottleneck when a higher volume of transactions is received by the system. Furthermore, the blockchain node provides perimeter security by allowing service deployment closer to the stakeholder.

A. Key Components of the Architecture

The proposed BaaS architecture is designed to offer various services for different stakeholders in the L5GO ecosystem. Here we propose a modularized service architecture. The BaaS architecture consists of different blockchain-based functions which are similar to network functions (NFs) in 5G networks. In contrast to the typical NFs in 5G, these blockchain-based functions are implemented on top of the blockchain by utilizing smart contracts. Then, these blockchain-based functions can be combined together to implement different blockchain-based services. These blockchain-based services are able to provide meaningful services for the stakeholders in the L5GO ecosystem. Multiple blockchain-based functions have to cooperate together to deploy each blockchain-based service. The operation of these functions and designed services can be customized according to the requirement and characteristics of the stakeholder.

1) Stakeholders: The proposed BaaS architecture is designed to provide services for different stakeholders in an L5GO ecosystem. Here we list all the stakeholders who are interacting in L5GO networks.

- **L5GOs:** This is the main stakeholder of the ecosystem, participating in all the services discussed in Section VI. The proposed BaaS architecture can support multiple L5GOs and support coordination among them.
- **MNOs:** One of the mobile service providers in roaming and offloading domains. Also, the operators who are willing to sell their own spectrum in the marketplace.
- **Mobile subscribers:** End users who receive mobile network services.
- **VNF vendors:** The companies who trade VNF as a service.
- **IoT data sellers/Tenants:** L5GOs who sell the collected IoT data.
- **Third Party Buyers:** Entities who intended to purchase the resources that are advertised in the marketplace.
- **Cloud service providers:** Vendors who fulfill the storage requirements of the system.

2) Functions: The BaaS architecture supports the modularized approach by defining a series of blockchain-based functions. These functions comprise the main building blocks of blockchain-based services enabled by the proposed architecture. The key blockchain-based functions supported by BaaS are as follows.

- **Subscription Management Function (SMF):** Manage the registration of the stakeholders and service applications.
- **Marketplace Function (MF):** Accommodate buying and selling services such as spectrum, VNFs, and IoT data.
- **Reputation Management Function (RMF):** Maintain the service quality of the system.
- **Selection Function (SF):** Execute selection strategies for picking optimal network providers (both roaming and offloading domains) and subscribers (offloading domain).
- **Fraud Prevention Function (FPF):** Enforce measures to avoid the occurrence of roaming frauds.

- **Data Management Function (DMF):** Provide IoT data storage and access solutions.
- **Agreement establishment and Payment settlement Function (APF):** Facilitate dynamic agreement negotiation and allow secure money transfer.

More details with respect to functions and implementation are presented in Section V.

3) Services: The BaaS architecture can be used to deploy different blockchain-based services for the L5GO stakeholders. The initiation of a blockchain-based service in BaaS architecture is done by combining the previously defined blockchain-based functions diversely. The functions defined above must be concatenated to a certain degree to deploy each blockchain-based service. Here, we list the some of the most important blockchain-based services which can be deployed by using the the previously defined blockchain-based functions.

- **Roaming Service:** Enable efficient roaming between MNOs and L5GOs.
- **Offload Service:** Facilitate efficient network load balancing.
- **Spectrum Sharing Service:** Accommodate spectrum trading between MNOs and L5GOs.
- **VNF Management Service:** Empower VNF resource trading between VNF vendors and L5GOs.
- **Identity Management Service:** Carry out stakeholder and resource registration operations.
- **IoT Data Management Service:** Enable L5GOs to share IoT data with third-party services.

More functional and implementation details about the services described above are presented in Section VI.

V. Key Functions of BAAS Architecture

The BaaS platform is a modularized architecture that comprises several blockchain-based functions. These functions behave as modules, which enables service providers to assemble them based on their diversified requirements and then to produce services. Some of such services are proposed in Section VI. These functions are necessarily structured to address the previously presented potential challenges in an L5GO ecosystem in Section IV. We have proposed seven such functions, and their respective operations are coded in the Ethereum smart contracts. The required services can be invoked by calling one or many functions sequentially, depending upon the requirements. The final outcomes of these combined functions—known as services—are explicitly explained in the next section.

The fundamental phases of the proposed functions are depicted in Fig. 3. The rest of the section presents the internal operation of the proposed BaaS functions. Table III depicts the summary of notations used throughout this section.

1) Subscription Management Function (SMF): The very first step is the registration of the stakeholders and service applications with the system. The service management function is proposed to serve the registration purpose. It allows the system to register details of stakeholders and various application-associated resources to the blockchain by the following steps.

TABLE III: Summary of notations

Notation	Description
$\text{Bandwidth}_{\text{Available}}$	Available Bandwidth
$\text{Bandwidth}_{\text{SystemMaximum}}$	Maximum Bandwidth of the System
C_i, C_P	i^{th} Cost, Product Cost
$\text{Capacity}_{\text{Available}}$	Available Capacity
$\text{Capacity}_{\text{SystemMaximum}}$	Maximum Capacity of the System
$\text{Cost}_{\text{Actual}}$	Actual Cost
$\text{Cost}_{\text{MaxSystem}}$	Maximum Cost of the System
D_{A_i}, D_{D_i}	i^{th} Advertised Data, i^{th} Deviation Data
D_i, D_{T_i}	i^{th} Data, i^{th} True Data
J_A, J_D, J_S	Allowed Jitter, Jitter Deviation, Session Jitter
L_A, L_D, L_S	Allowed Latency, Latency Deviation, Session Latency
P_{B_A}	Allowed Blocking Probability
P_{B_D}	Blocking Probability Deviation
P_{B_S}	Session Blocking Probability
PL_A	Allowed Packet Loss
PL_D	Packet Loss Deviation
PL_S	Session Packet Loss
R_P	Network Provider's Reputation Score
$R_{PMA\ new}$	New Moving Average of the Reputation Score of a Network Provider
$R_{PMA\ old}$	Old Moving Average of the Reputation Score of a Network Provider
R_S	Seller's Reputation Score
$R_{SMA\ new}$	New Moving Average of the Reputation Score of a Seller
$R_{SMA\ old}$	Old Moving Average of the Reputation Score of a Seller
S_C	Cost Score
S_O, S_R	Offloading Score, Roaming Score
SRF	Seller Rating Factor
$\text{SS}_{\text{Available}}$	Available Signal Strength
$\text{SS}_{\text{SystemMaximum}}$	System's Maximum Signal Strength
W_{C_i}	Weight of i^{th} Cost
W_{C_P}	Product Cost Weight
W_{D_i}	Weight of i^{th} Data
W_{DD_i}	Weight of i^{th} Deviation Data
W_J, W_L	Jitter Weight, Latency Weight
W_{P_B}	Blocking Probability Weight
W_{PL}	Packet Loss Weight
W_{R_S}	Seller's Reputation Weight

Step 1: MNOs or L5GOs can record each stakeholder or resource details. This information stores off-chain and adds the hash of the registry data structure in the distributed ledger. Here different stakeholders need to provide different information during the registration. Table IV presents a list of parameters of each user that can be collected during the registration process. Some of this information, such as resource information, can be changed dynamically.

Step 2: Next, the blockchain assigns a unique ID and a universal wallet to each user.

User verification is also handled under this module as follows.

Step A: User sends a request for access along with their universal identity, whenever the user on-boards to the L5GO network.

Step B: Consequently, the edge node searches for the stored hash value for the corresponding received ID from the distributed ledger and hashes the received

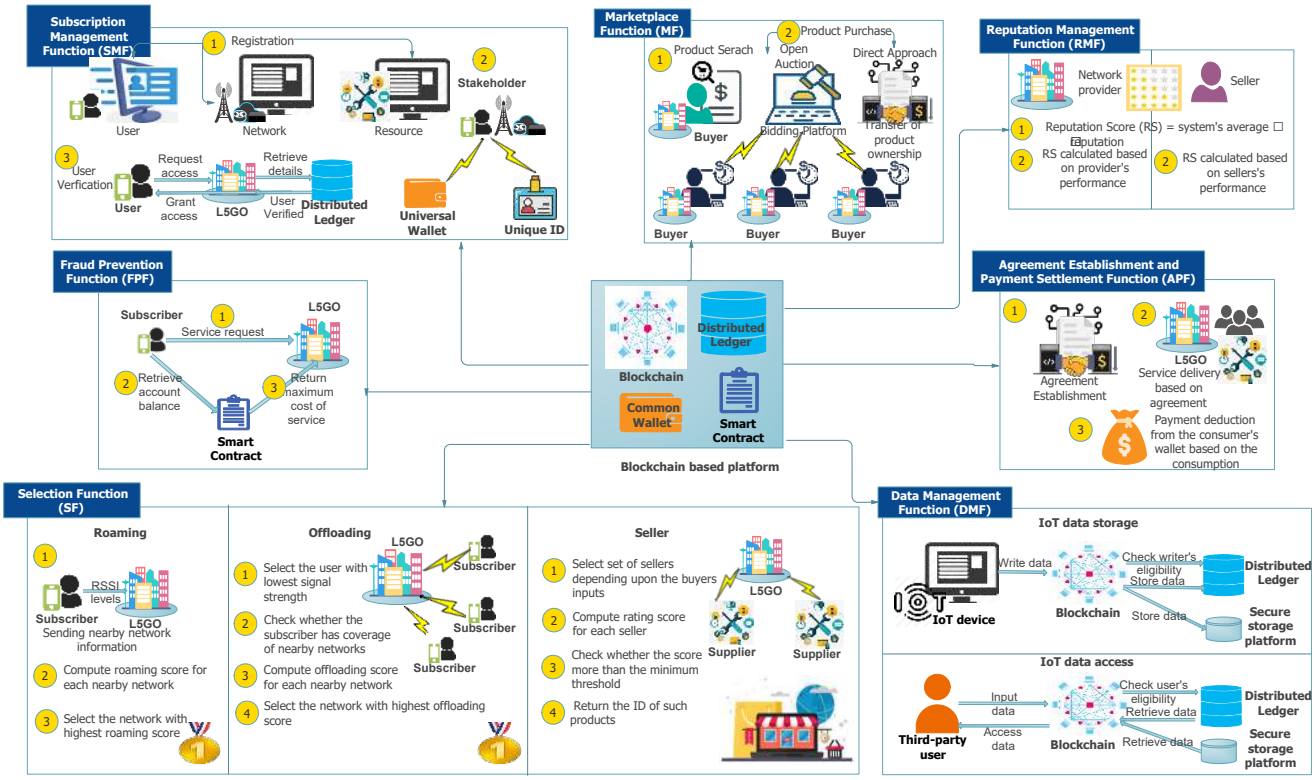


Fig. 3: Key functions of the proposed BaaS architecture.

user information.

Step C: Then, the edge node will grant access if the stored and received hash values are the same.

TABLE IV: Registration details.

	Types	Parameters
Stakeholders	MNOs, L5GOs	Id, Network bandwidth, Network capacity, Charging scheme
	Subscribers, IoT device owners, Sellers, Buyers	Id, Name, Social security number, Home address
Resources	Spectrum	Id, Price, Detailed Description, Leasing period, GPS location, Owner's address, Band range, Channel Quality (SNR)
	VNFs	Id, Price, Detailed Description, Leasing period, GPS location, Owner's address, VNF type, VNF developer, Memory, Disk space, CPU cores
	IoT data	Id, Price, Detailed Description, Leasing period, GPS location, Owner's address, Data source URL, Data stream type, Company name
	IoT devices	Id, Owner address

2) Marketplace Function (MF): Marketplace function is proposed to create a platform for sellers to advertise their products and for customers to purchase products conveniently. Different section algorithms and bidding mechanisms can be integrated with this function via smart contracts for selecting the best available product. Use of smart contracts can be further utilized to automate the selection process. The step-by-step process for automatic selection of a product is explained below.

Step 1: Buyer inputs the purchasing information such as the leasing period, GPS location, expected rating, etc.

Step 2: Next, a Seller Rating Factor (SRF) is calculated for each seller as below; this rating factor is used to select the suitable seller for each buyer request.

$$SRF = R_{\mathrm{S}} * W_{\mathrm{R_S}} / C_{\mathrm{P}} * W_{\mathrm{C_P}} \quad (1)$$

Step 3: Subsequently, a seller is selected for each buyer based on the following condition:
If the condition (Minimum Rating Threshold [MRT] $<$ SRF) is true, then MRT is updated with the SRF value and returns the ID of the particular product.

However, product purchase can be done in two ways. Namely, direct purchasing and open auction. The below mentioned steps should be followed to purchase a product directly from the seller.

Step L4: Initially, a buyer inputs the ID of the product.

Step L5: System checks the availability of the product and whether the buyer has enough cash.

Step L6: Transfer the product ownership to the buyer.

Step L7: Buyer pay the seller by sending payment.

The open auctioning method is implemented as follows:

Step R4: Selected sellers start the bidding process to sell the product.

Step R5: Buyer or buyers who are willing to buy this product start bidding within the advertised time period. The system selects the highest bidder and reserves the bid amount from their wallet.

Step R6: If the highest bid is raised, the second highest bidder will receive their reserved bid back.

Step R7: When the bidding time expires, the contract transfers the money (highest bid) to the seller.

The entire marketplace process is depicted in Fig. 4.

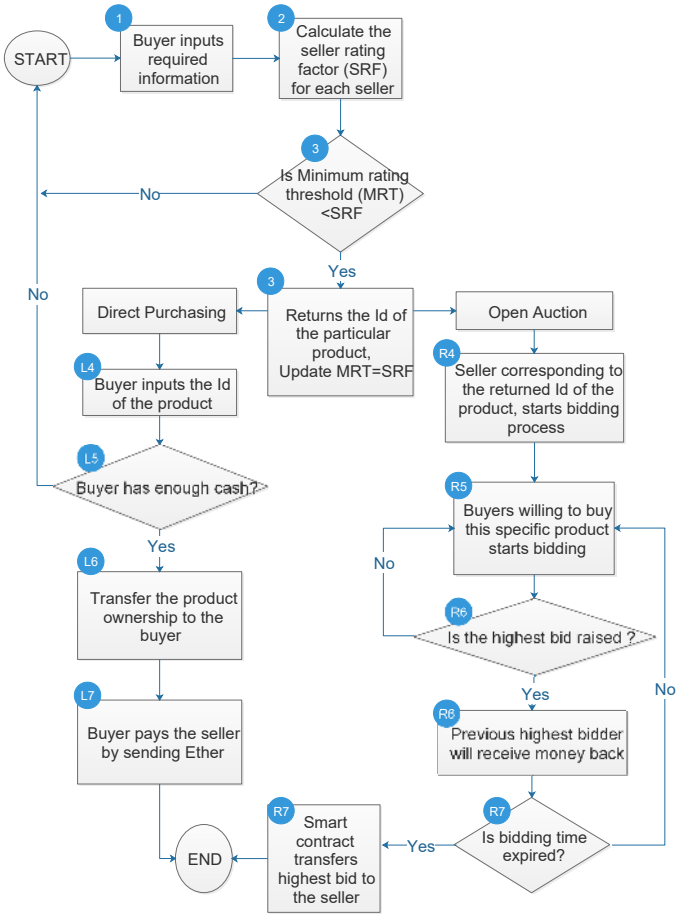


Fig. 4: The flow of marketplace function.

3) Reputation Management Function (RMF): Our system evaluates the quality of services offered by the different stakeholders. Such historical performance information will be utilized to prioritize the stakeholders and define the payment rates for their services. Therefore, we propose a novel reputation management function to evaluate the products and services offered by the network providers. Mainly, this reputation management function calculates the reputation scores for each of the network providers that can mainly be used for roaming and offloading services. It is also used for reviewing the sellers associated with the marketplace.

The steps below are followed to calculate the reputation score for the network provider during the roaming and offloading events.

Step R1: Initially, reputation scores of all the network providers are set on the system's average reputation and then updated gradually.

Step R2: Next, the reputation score is calculated at the end of each successful session based on the following performance characteristics: latency, packet loss, jitter, and blocking probability. Firstly, for each of these parameters, a normalized deviation is calculated as follows.

$$L_{\mathrm{D}} = \frac{L_{\mathrm{A}} - L_{\mathrm{S}}}{L_{\mathrm{A}}} \tag{2}$$

$$PL_{\mathrm{D}} = \frac{PL_{\mathrm{A}} - PL_{\mathrm{S}}}{PL_{\mathrm{A}}} \tag{3}$$

$$J_{\mathrm{D}} = \frac{J_{\mathrm{A}} - J_{\mathrm{S}}}{J_{\mathrm{A}}} \tag{4}$$

$$P_{\mathrm{B_D}} = \frac{P_{\mathrm{B_A}} - P_{\mathrm{B_S}}}{P_{\mathrm{B_A}}} \tag{5}$$

$$R_{\mathrm{P}} = W_{\mathrm{L}} * L_{\mathrm{D}} + W_{\mathrm{PL}} * PL_{\mathrm{D}} + W_{\mathrm{J}} * J_{\mathrm{D}} + W_{\mathrm{P_B}} * P_{\mathrm{B_D}} \tag{6}$$

Here, the values for the weights can be updated according to the polices defined by the system. The sum of all weight values are equal to 1. Please note that higher the deviation values means that particular session had a better performance.

Step R3: Finally, the moving average of the reputation score is calculated as below.

$$R_{\mathrm{PMAnew}} = \alpha * R_{\mathrm{P}} + \beta * R_{\mathrm{PMAold}} \tag{7}$$

Note: α + β = 1

Here, the values for the weights (i.e. α, β) can be updated according to the polices defined by the system.

The following steps are used to compute the reputation score of the seller for market place related events.

Step L1: Initially, reputation scores of all the sellers are set on the system's average reputation and then updated gradually.

Step L2: Next, the reputation score of a seller is calculated as follows:

$$D_{\mathrm{D_i}} = (D_{\mathrm{T_i}} - D_{\mathrm{A_i}})/D_{\mathrm{A_i}} \tag{8}$$

$$R_{\mathrm{S}} = \sum_{i=1}^{n} D_{\mathrm{D_i}} * W_{\mathrm{D_i}} \tag{9}$$

Here, The sum of all weight values are equal to 1. Moreover, the values for the weights can be updated according to the polices defined by the system.

Step L3: Finally, the moving average of the reputation score is calculated as below.

$$R_{\mathrm{SMAnew}} = \alpha * R_{\mathrm{S}} + \beta * R_{\mathrm{SMAold}} \tag{10}$$

Note: α + β = 1
Here also, the values for the weights (i.e., α, β) can be updated according to the polices defined by the system.

The reputation system of the proposed model is shown in Fig. 5.

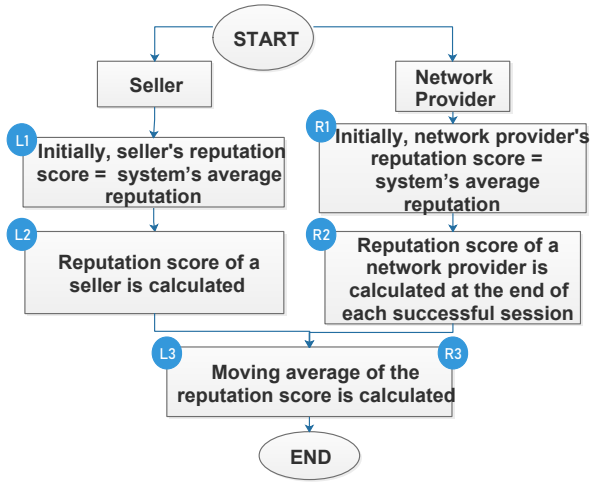


Fig. 5: The flow of reputation management service.

4) Selection Function (SF): The system has to do different selections tasks such as selecting the best L5GOs to perform an offload or roaming task. Therefore, we propose a selection function that allows the system to automatically select the optimal network provider for a mobile user performing a roaming and offloading event. During offload events, MNOs have to select the optimal subscriber or subscribers to offload. Thus, the selection function can also decide the optimal subscriber to offload.

The approach outlined below is used to find the best available L5GO to the subscriber while roaming.

Step L1: User sends a connection request and details of k number of nearby networks to a nearby L5GO.

Step L2: Subsequently, a roaming score is computed for each network provider as follows,

$$S_{\mathrm{R}} = \sum_{i=1}^{3} (D_{\mathrm{i}} * W_{\mathrm{D_i}}) \tag{11}$$

Note: D_1 = normalized available signal strength (i.e. $D_1 = SS_{\mathrm{Available}}/SS_{\mathrm{SystemMaximum}}$, here SS = Signal Strength), D_2 = reputation score(From equation 7), D_3 = cost score (From equation 12). Moreover, The sum of all weight values are equal to 1.
Cost score can be calculated by using equation 12

$$S_{\mathrm{C}} = \sum_{i=1}^{3} (C_{\mathrm{i}} * W_{\mathrm{C_i}}) \tag{12}$$

Note: C_1= Normalized cost for voice, C_2 = Normalized cost for SMS, C_3 = Normalized cost for data. The sum of all weight values are equal to 1 and the values for the weights can be updated according to the polices defined by the system. Normalized costs for each service is calculated by using equation 13.

$$C_{\mathrm{i}} = \frac{Cost_{\mathrm{MaxSystem_i}} - Cost_{\mathrm{Actual_i}}}{Cost_{\mathrm{MaxSystem_i}}} \tag{13}$$

$Cost_{\mathrm{MaxSystem_i}}$ is the maximum asking cost by any user in the system

Step L3: Then, the L5GO with the highest roaming score out of all the registered networks is selected

The process of selecting a network provider during the offloading is as follows,

Step M1: MNO acquires list of available networks for a selected subscriber

Step M2: Subsequently, an offloading score is computed for each network provider as below,

$$S_{\mathrm{O}} = \sum_{i=1}^{4} (D_{\mathrm{i}} * W_{\mathrm{D_i}}) \tag{14}$$

Note: D_1 = Normalized available capacity (i.e. $D_1 = Capacity_{\mathrm{Available}}/Capacity_{\mathrm{SystemMaximum}}$),
D_2 = Normalized network bandwidth (i.e. $D_2 = Bandwidth_{\mathrm{Available}}/Bandwidth_{\mathrm{SystemMaximum}}$),
D_3= Cost score (From equation 12),
D_4= Reputation score (From equation 7).
Moreover, the sum of all weight values are equal to 1 and the values for the weights can be updated according to the polices defined by the system.

Step M3: Then, the L5GO with the highest offloading score out of all the registered networks is selected

The process of selecting the most eligible subscriber to offload to an L5GO network is outlined below,

Step R1: if the HMNO's capacity utilization is higher than a pre-defined threshold value of the total capacity, operator selects a subscriber connected with the least signal strength

Step R2: Then, checks whether the chosen user has coverage of any other nearby networks

Step R3: If the above condition is satisfied, system outputs the ID of the selected subscriber. If it is not, the system will repeat the same procedure for the next user connected with lowest signal strength

A flow chart for the above explained selection processes is demonstrated in the Fig. 6.

5) Fraud Prevention Function (FPF): Fraud prevention function is defined to eliminate the impact of fraud during roaming and offloading events. Specifically, it focuses on preventing the over utilization of resources by visiting users.

- Whenever a mobile subscriber requests a service from the visitor L5GO, the system will check whether the subscriber has enough credits in his/her wallet
- If the above condition is true, system will calculate the maximum cost for service that L5GO can charge the subscriber, based on customer's remaining account balance and the percentage of MNO's revenue agreed to pay for the L5GO for its delivered service

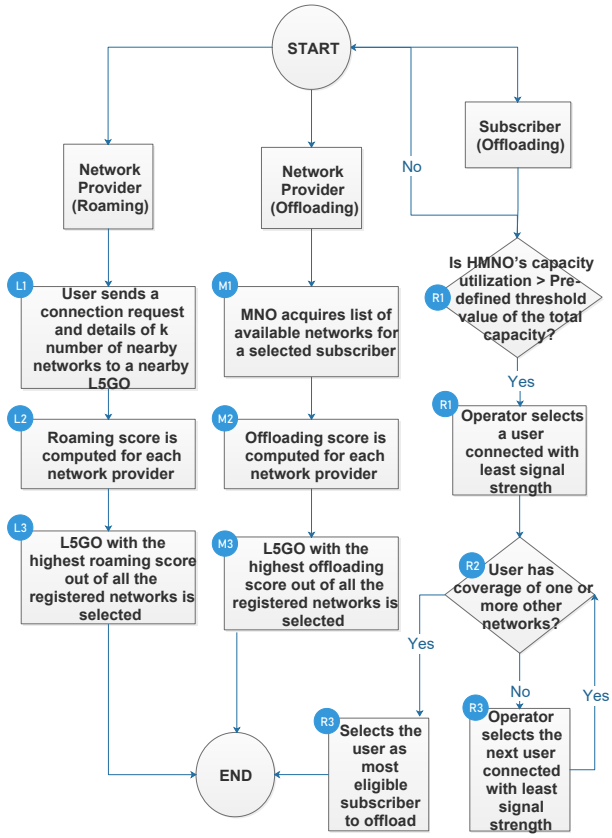


Fig. 6: The flow of selection function.

- Then, VPMN provides the service only up to the calculated threshold amount. Therefore, no subscriber is able to over-utilize the assigned spectrum.

6) Data Management Function (DMF): L5GO networks usually consist of various IoT devices. The collected IoT can be shared with other users. We propose a data management function mainly focused on two major aspects in IoT data management, i.e., data storage and data access management.

The IoT data storage process is handled as follows,

Step L1: Initially, devices write data to the blockchain by providing the owner's address and device ID

Step L2: Then, the responsible smart contract checks if the owner's address corresponds to the device ID

Step L3: Subsequently, store the hash of the data in the blockchain and store the original data in a secure storage platform (off-chain) or in the distributed ledger (on-chain)

IoT data access process is managed as below,

Step R1: Initially, specific third-party user inputs the device owner's address and the device ID to the blockchain platform

Step R2: Then, the eligibility of the third-party user is verified by checking whether the device owner has given the access permission

Step R3: If the access is granted, the hash of the data is returned and used to retrieve data from the storage platform or the distributed ledger

The proposed IoT data management functions are briefly explained Fig. 7.

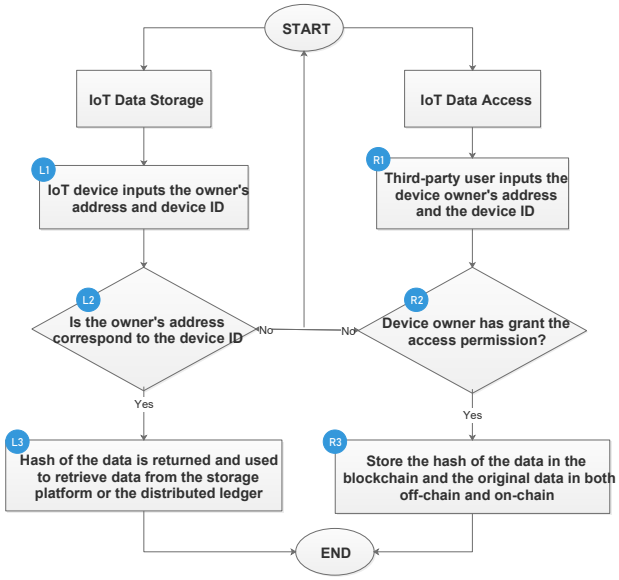


Fig. 7: The flow of IoT data management function

7) Agreement establishment and Payment settlement Function (APF) : Most of the blockchain-based services related to L5GO networks involve the establishment of dynamic agreements between different stakeholders and settling payments for services. The APF function is proposed to offer these services. This service is offered for all the stakeholders in the system as given below:

- Dynamic agreement is established whenever an optimal network provider or a seller is selected for a subscriber or a buyer, respectively
- If the subscriber or the buyer requests a service from the visitor L5GO or the seller, respectively, a specific smart contract will execute and check whether the subscriber or the buyer has enough cash
- Payments are deducted directly from the subscriber's or the buyer's wallet based on agreed policies

VI. Proposed Services in BaaS

We propose a novel method of deploying blockchain-based services in L5GOs. The proposed BaaS platform is a modularized architecture, which enables combination of previously defined functions in section V, and then to produce different services. In this section, we explicitly illustrate the method of combining these defined components to offer numerous services related to L5GOs. These blockchain-based services and their features are depicted in Fig. 8.

A. Roaming Service

In the BaaS platform, the roaming service can be implemented with the five previously defined modules as given below. Details of subscribers and network providers are recorded

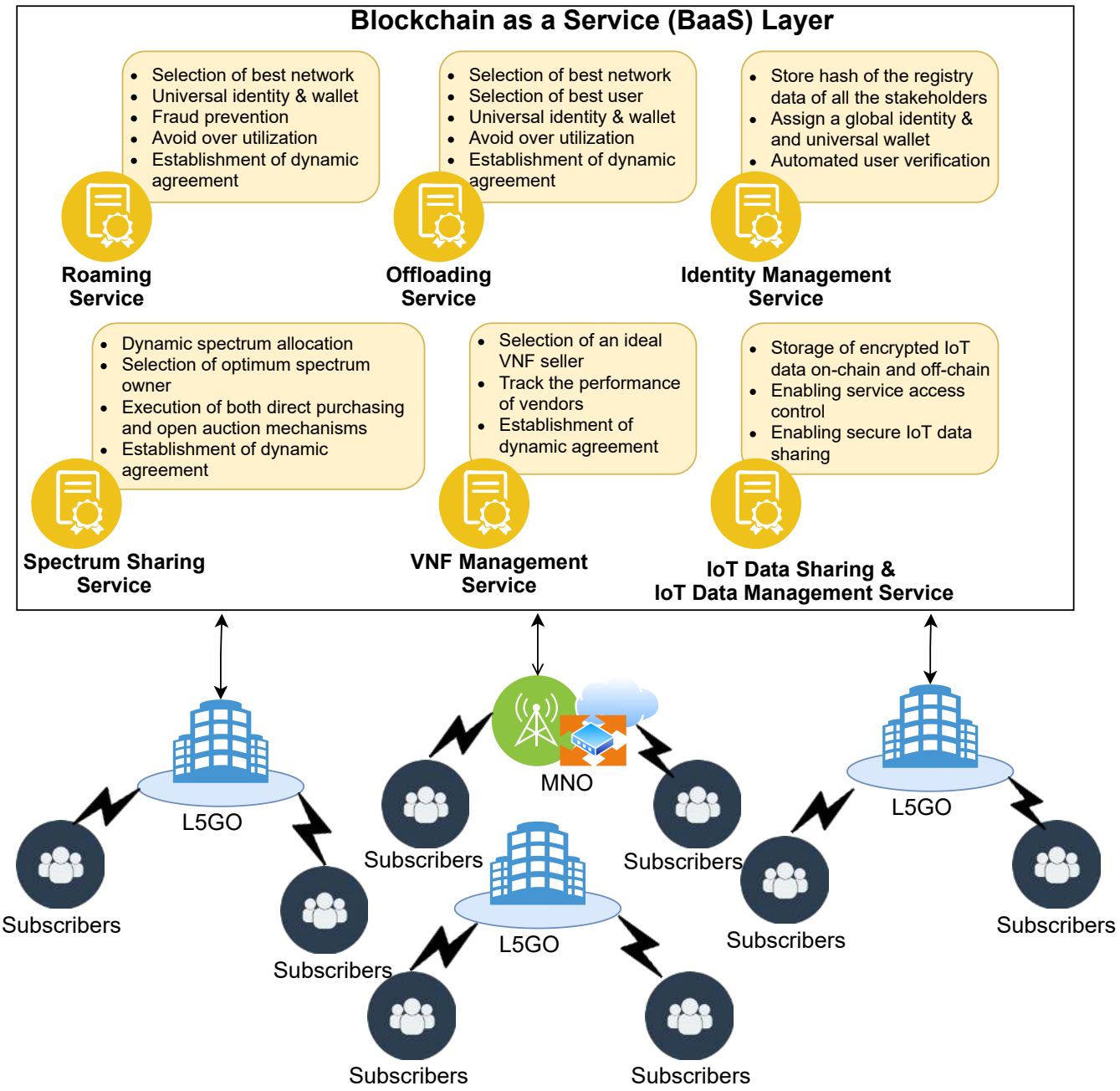
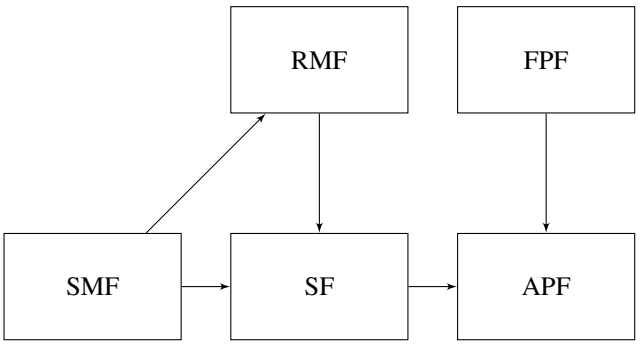


Fig. 8: Services offered by Blockchain.

under the subscription management component. Whenever the roaming user sends an access request to the nearby L5GO, the user verification process will be initiated, which is also handled by the previously mentioned component. Then, the selection module will select the optimal network provider based on the reputation, charging scheme, and the signal signal strengths of nearby networks. The reputation for each L5GO is calculated under the reputation management component. Subsequently, the subscriber will be offloaded to the selected L5GO. Finally, agreement establishment and payment settlement between stakeholders will be handled as described under the APF component. Also, the fraud prevention module is added to the structure to avoid over-utilization of VPMN's resources.

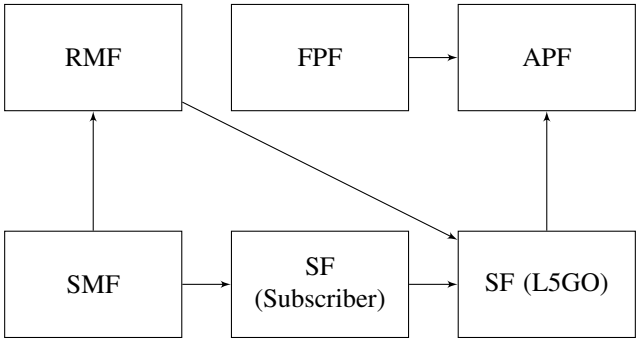


In contrast to the static roaming agreements, the proposed roaming mechanism supports the establishment of dynamic roaming agreements based on reputation score. MNOs and

L5GOs have the flexibility to prioritize the selection parameters by changing the weights in section function (see equation 12). Moreover, this reputation score can be used to adjust the payment for offered roaming services which will motivate visitor network operators to offer high-quality roaming sessions. In addition, the proposed roaming service eliminates involvement of third-party clearing houses in traditional roaming process and prevents over-utilization of VPMN's resources.

B. Offload Service

The block diagram for the offload process depicted below has similar functionalities as the roaming process except for the subscriber selection module. This module is used to check the eligibility of a subscriber to offload from a overloaded network provider.



The proposed offload mechanism offers the flexibility for MNOs and L5GOs to prioritize the selection parameters by changing the weights in section function (see equation 14). Moreover, this reputation score can be used to adjust the payment for offered services by the offloaded networks. This will motivate visitor network operators to offer high-quality services for offloaded customers. Moreover, the proposed offloading mechanism also supports the establishment of dynamic roaming agreements, in contrast to the static roaming agreements in Traditional System (TS).

C. Spectrum Sharing Service

The spectrum-sharing service comprises five defined components. The subscription management module logs necessary details of spectrum sellers and buyers and carries out the stakeholder verification process. Then, the buyers initiate the process of searching the required spectrum via the marketplace module. Next, the system selects the optimal seller for the buyer based on sellers' reputations and charging schemes through the selection component. Finally, the procedure to purchase the spectrum is mentioned in the Marketplace component. Required inputs to the searching process under the marketplace module and inputs to the reputation measurements of the seller under the reputation module are listed in Table V.

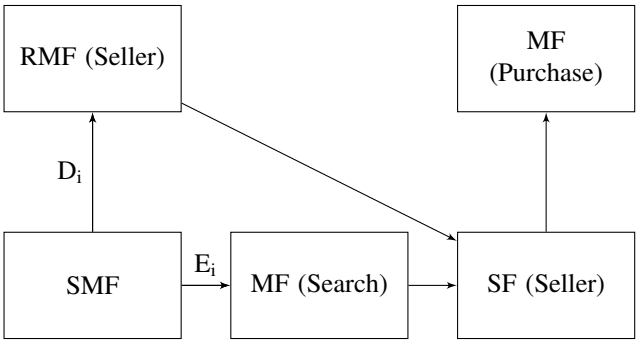


TABLE V: Input parameters associated with the spectrum sharing service.

Inputs	Parameters
E_i	E_1= Band range,E_2= Leasing period, E_3=GPS location
D_i	D_1= Channel quality, D_2= Leasing period

The limitations on catering to the demands of 5G networks with the utilization of traditional static spectrum allocation methods are resolved in the proposed system with the execution of dynamic spectrum sharing solutions. In addition, the current payment system is automated with the blockchain. Moreover, the optimal spectrum sharing partner is selected for each network provider based on their requirements, which would ease the current selection process. Additionally, mutual trust and secure transaction are ensured between trust-less entities. Furthermore, single point failures are eliminated by deploying the centralized services on a decentralized setup with the incorporation of smart contracts.

D. VNF Management Service

The structure of the VNF management service is the same as the block arrangement of the spectrum sharing service, except for the D_i and E_i module inputs. These input data are recorded in Table VI.

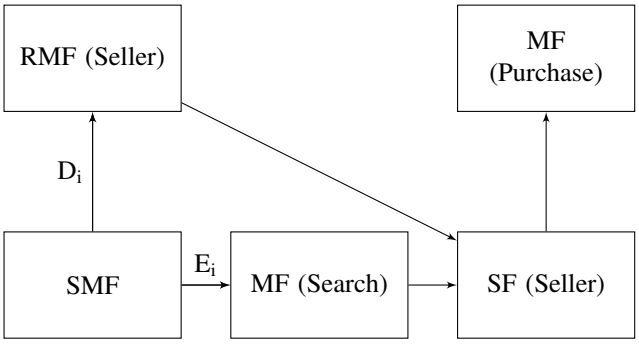


TABLE VI: Input parameters associated with VNF management service.

Inputs	Parameters
E_i	E_1= VNF type, E_2= VNF developer, E_3= Leasing period, C_4=GPS location
D_i	D_1= Memory, D_2= Disk space, D_3= CPU cores, D_3= Leasing period

Replacement of traditional third-party brokers in a resource management platform using our proposed solution for VNF

management produces many advantages. It can cut down on extra expenses and unnecessary delay components and assure secure and trusted VNF trading among multi-operators by enabling transactions via smart contracts. Additionally, selection of an ideal VNF seller based on reputation and cost factors is offered, which would mainly help the new tenants in choosing the best matching seller among anonymous traders. Moreover, the current quality in providing VNF services are improved by triggering competition among service providers with the execution of a reputation management system.

E. IoT Data Sharing Service

IoT devices record beneficial data that could be shared between interested parties and could be sold in a marketplace platform. We propose the same block arrangement of the spectrum-sharing service to the IoT data sharing service, since both the approaches center around the marketplace concept. However, the inputs to the modules vary, as shown in the below diagram. Table VII lists the essential inputs to the marketplace (search) and reputation (seller) components.

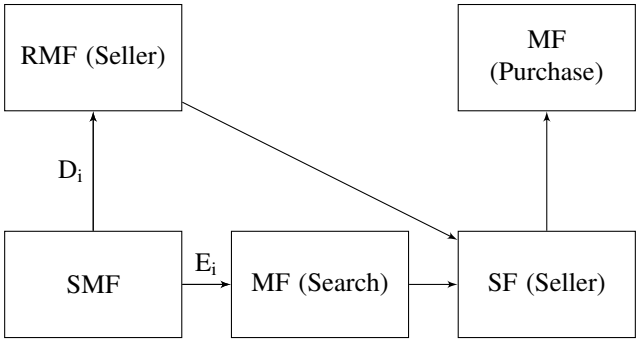


TABLE VII: Input parameters associated with IoT data sharing service

Inputs	Parameters
E_i	E_1= Data stream type, E_2= Company name
D_i	D_1= Leasing period

The proposed architecture facilitates use case–specific distributed IoT data-sharing operations utilizing the blockchain technology, in contrast to current centralized systems. Thus, the IoT data acquired from different industries will be shared securely to the necessary parties upon authorization. Furthermore, the proposed scheme executes dynamic and transparent agreements instead of static agreements when trading IoT data against a compensation to speed up the sharing process. Moreover, the current manual payment procedures are automated in the proposed scheme by enabling dynamic payment systems built-in with blockchain, which accelerates the payment process. Additionally, the system operates as a decentralized marketplace operated by smart contracts to incorporate multiple parties to open bids for the IoT data for purchasing, which ensures the fairness of the system compared to the TS.

F. Identity Management Service

Subscription management is the only block required to represent the identity management service.

The proposed scheme avoids identity theft, which is one of the major hurdles in current subscription management platforms, by hiding registry data of stakeholders with the use of an encryption algorithm. Furthermore, multiple registration times at different checkpoints in the same platform are avoided with the assignment of a unique ID to each stakeholder.

G. IoT Data Management Service

The IoT data management service consists of three modules as shown in the block arrangement below. Initially, IoT devices and their owner details are logged via the subscription management component. Then, the IoT data storage and IoT data access approaches are handled through the data management block. Finally, dynamic agreement establishment and payment settlements between selected parties are managed through the APF module.

The proposed architecture advances the management of the IoT data process by leveraging the distributed ledger based decentralized service architecture. Furthermore, the current systems transfer sensitive IoT data to third party service providers for the data storage to eliminate capacity overflows in IoT devices, which will eventually create privacy issues. This challenge is addressed in the proposed system with the utilization of hashing algorithms when storing data in the distributed ledger to ensure integrity. Additionally, data access permissions are only granted to the authorized parties whereas the state of art systems are lacking in such a formal authentication mechanism to control access.

VII. Numerical Analysis

We conducted various simulations to evaluate the performance of the proposed BaaS architecture. This section presents the simulation models generated using Matlab [36] and the obtained simulation results. These tests are mainly carried out to provide a comparison with existing systems and identify the benefits of proposed blockchain-based approaches.

Fig. 9 shows the experimental model that we used to analyze the proposed three simulation models—namely, roaming cost, roaming service quality, and reputation management of VNF deployment. Based on Fig. 9, the simulation model consists of one hundred devices, ten L5GOs, and 10 VNF providers. However, the interaction between stakeholders varies depending upon the situation. For instance, with reference to simulation model 1, only a connection between a user and an L5GO is considered. In contrast, simulation model 2 considers 10

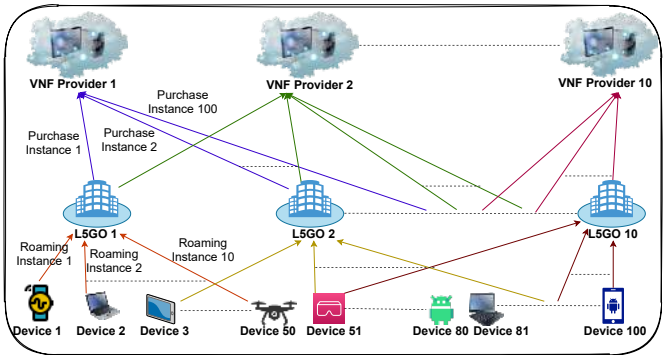


Fig. 9: Experimental System model

L5GOs, and each of them is connected to 10 users, making a total of 100 roaming instances. Simulation model 3 considers 10 VNF vendors and each of them provides services to 100 L5GOs, making the total purchase instances to 1000.

The rest of the section explicitly discusses the experimental setup, methodology, and results along with the results representation.

A. Simulation Model 1: Roaming Cost

A cost analysis is carried out to analyze the charges involved when delivering roaming services via traditional and proposed blockchain-based systems. In this experiment, we consider the charges for broadband service during the roaming. That is, the consumer charges per unit MB. Initially, necessary equations are formulated and the utilized notations in these formulations are listed in Table VIII. Similarly, the model can be used for voice call services as well.

TABLE VIII: General Simulation Parameters.

Notation	Description	Value
C_B	Blockchain-based cost	€2.40 (table XII)
C_C	Current system based roaming charge per session duration	
C_{CT}	Current system based roaming charge per session duration with tax	
C_{DCH}	Cost for DCH	
C_F	Cost for fraud	
C_{IC}	Cost for international carrier	€0.02 [37]
C_P	Proposed system based roaming charge per session duration	
C_{PT}	Proposed system based roaming charge per session duration with tax	
C_{RD}	Cost for research and development	
C_U	Cost of the unit	€0.0034 [38]
E_M	Expected margin	
P_{DCH}	DCH percentage	1% [39]
P_E	Expected margin percentage	8% [40]
P_F	Fraud percentage	5% [41]
P_{RD}	Research and development percentage	2% [42]
P_T	Tax percentage	10.24% [43]
R_M	Revenue of a mobile operator	
S_D	Session duration	

In traditional mobile systems, the roaming charge depends on the cost of the data unit, cost for the international carrier, expected margin, payment for the third-party, double taxation, and other investments such as those for research and development. Equation 15 represents roaming charges per session duration for the current system, which is a combination of the aforementioned factors.

$$C_C = (C_U + C_{IC} + E_M + C_{DCH} + C_F + C_{RD}) * S_D \quad (15)$$

The total cost of a roaming subscriber will includes a tax, which will be imposed on Equation 15.

$$C_{CT} = C_C + C_C * P_T \quad (16)$$

The roaming charges per session for the proposed blockchain-based data roaming services are expressed in Equations 17 (excluding tax) and 18 (including tax).

$$C_P = C_B + (C_U + C_{IC}) * S_D + C_{RD} \quad (17)$$

$$C_{PT} = C_P + C_P * P_T \quad (18)$$

To realize the above four formulations, computation of the following equations are necessary—primarily, the revenue of the mobile operator, which is given in Equation 19.

$$R_M = C_U + C_{IC} \quad (19)$$

Different percentages of the operator's income are utilized for several functions, such as expected margin value, payments for DCH, fraud, and other investments as shown in Equations 20, 21, 22 and 23 respectively.

$$E_M = R_M * P_E \quad (20)$$

$$C_{DCH} = R_M * P_{DCH} \quad (21)$$

$$C_F = R_M * P_F \quad (22)$$

$$C_{RD} = R_M * P_{RD} \quad (23)$$

Based on Equations 16 and 18, the roaming cost per session for both traditional and proposed roaming systems are calculated by varying the session duration from 1 GB to 10 GB. The final outcomes of this experiment are presented in Fig. 10.

Based on Fig. 10, the blockchain approach is expensive compared to the TS only at the initial stage. This is because the extra cost is incurred for the smart contract deployment and it is only a one-time operation. Our solution is cheaper than the current system for longer sessions, since no additional payments are expended for third-party service delivery and alternative fraud prevention systems. Therefore, the execution of a blockchain-based system is cost effective for longer sessions.

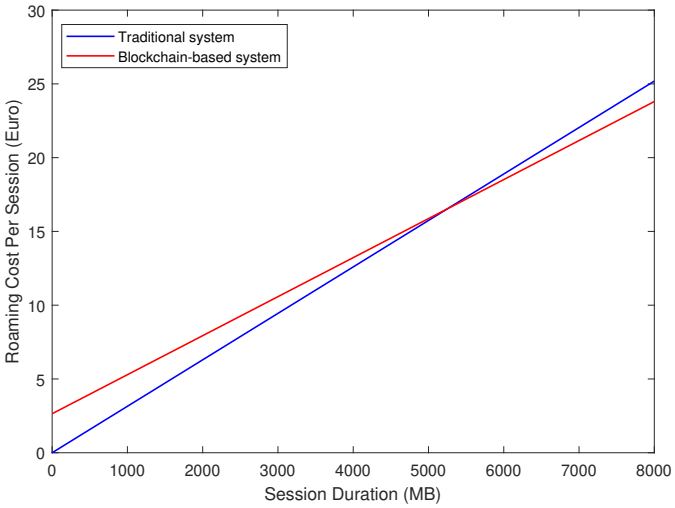


Fig. 10: Cost comparison between traditional and proposed model.

B. Simulation Model 2: Received Service Quality for Roaming Users

A test is carried out to compare the received service quality of roaming users. For this, a system model of 10 L5GOs and 100 users is considered. In the TS, the HPMN allocates a VPMN for their own subscribers based on a static agreement. Hence, TS is not able to guarantee their users, a definite connectivity with the network provider who has the best quality of the network conditions. However, in our approach, the system selects a L5GO dynamically based on the nearby operator's signal strength, cost, and performance history.

In this experiment, the network selection process of the TS is modeled in a way where the TS chooses a random operator to offload its subscriber. On the other hand, the proposed system is developed to find the optimal operator by generating random values for signal strength, cost, and reputation scores within a reasonable range; i.e., session cost on the interval 0.9–1.1 with average of 1 Euro following a uniform distribution, signal strength values from the discrete uniform distribution on the interval 1 to 100 with average of 50, and reputation rating scores from 50 to 100 with average of 75.

In our model, the roaming selection scores (Equation 11) for ten operators are computed, and the network provider with the highest score is selected for each user. Subsequently, the same procedure is repeated for 100 subscribers. Four types of proposed systems are modeled by varying the prioritization factors, which are given in Table IX.

TABLE IX: Types of proposed systems.

Simulation Model	$W_{\text{Signal Strength}}$	W_{Cost}	$W_{\text{Reputation}}$
Proposed System 1 (PS1)	0.33	0.33	0.33
Proposed System 2 (PS2)	0.5	0.25	0.25
Proposed System 3 (PS3)	0.25	0.5	0.25
Proposed System 4 (PS4)	0	1	0

[a] Reference to equation 11

Please note that in our approach, operators with higher signal strength ratings offer more reliable connection, whereas the operators with higher reputation rating provide a better quality service. Moreover, the operators with the higher cost rating offer a cheaper service. The network selection algorithms for traditional and blockchain-based algorithms were run for this system model and the generated results are tabulated in Table X. Subsequently, these these experimental data are summarized in Table XI.

TABLE X: Numerical results.

Simulation Model	Signal Strength	Cost	Reputation
TS	51.01 ± 6.0804	100.0398 ±1.1856	75.04±2.9318
PS1	86.76 ±2.2415	101.6876 ±1.0904	88.82±1.8219
PS2	91.00 ± 1.7557	101.0131±1.1633	83.56±2.102
PS3	85.76 ±2.4377	103.5026±0.904	87.91±1.8847
PS4	54.29± 5.9829	91.5286±0.2927	74.43±3.1857

TABLE XI: Summary of simulation results.

Tested Parameter	Outcome
Signal Strength	PS2 > PS1 > PS3 > PS4 > TS
Cost	PS4 > PS3 > PS1 > PS2 > TS
Reputation	PS1 > PS3 > TS > PS2 > PS4

According to Table X, based on the obtained numerical results for the signal strength, it is clear that the proposed system outperformed the current system in all occasions where the decision is made by taking multiple factors into account. However, it is also evident that poor signal strengths are received when only the cost factor is considered when selecting the operator. The highest average signal strength is obtained when the weight of the signal strength is increased over other weights. When given equal weights for all factors, a slightly lower average signal strength is experienced.

Based on the acquired numerical data for the cost factor, the TS demonstrates a higher cost compared with all the proposed models, as it requires the additional cost for the execution of fraud prevention systems and to pay for the DCH for their delivered services.

With reference to the tabulated numerical results in Table X, for the reputation parameter, we can observe that our system picks the operator with excellent track records. This is because the reputation data are taken into consideration when calculating the network selection algorithm. The PS1 depicts best results amongst all other models since it gives more priority to the reputation factor.

Based on the outcomes of Table XI and considering the proposed models, it is noticeable that the system with the highest weights of a given evaluation factor surpasses the other systems.

C. Simulation Model 3: Impact of Reputation Management on VNF deployment

The proposed system considers the reputation of VNFs in its selection. The impact of the system's reputation with the error probability was experimented in Matlab.

Generally, operators are confined to a particular VNF seller. However, they can deliver poor service sometimes. Further,

operators pay the seller constant amount based on agreed conditions regardless of their poor service records. In our methodology, payments are made based on seller's reputation to excel their service quality, according to Equation 9 and 1.

In this experiment, a certain VNF seller is selected and 100 of its purchase instances are examined. Several graphs are generated in Fig. 11 varying error probability by 0%, 0.01%, 0.1%, 1% and 5% .

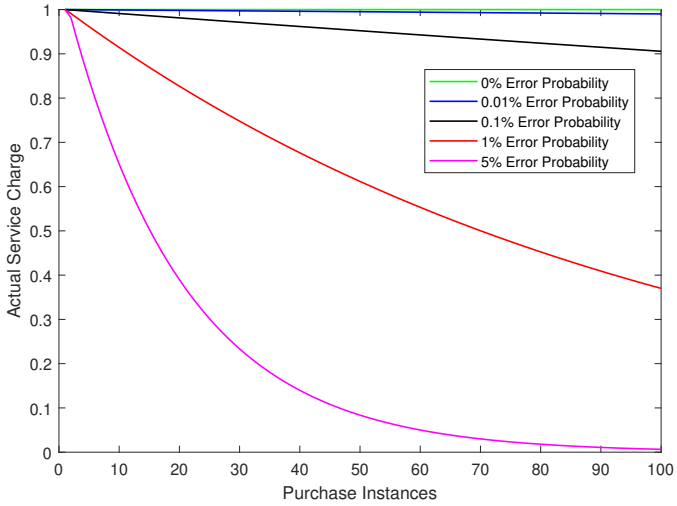


Fig. 11: Variation of payment for VNF service for different error probabilities.

Based on Fig. 11, the maximum reputation is reached when there is no deviation in the agreed service quality. Furthermore, the reputation score reduces largely with the increase of error probability, which makes the buyer pay a low service charge. This is due to the fact that the payment is directly proportional to the reputation. Therefore, the operator has to pay the reduced percentage of reputation of the advertised cost.

A comparison is carried out between the traditional and proposed models based on reputation and error rate, considering the VNF management application. For this, 10 VNF operators and 100 purchase instances per each VNF operator are considered. Since the operators with lesser reputation scores have the higher error rate, we define the instantaneous error rate as a function of their reputation score (Equation 24).

$$ErrorRate = GlobalErrorRate * (1 - R_S) \qquad (24)$$

The global error rate refers to the probability of an error to have occurred in the system. Initially, it is set to 0.1 and reputation scores of operators are randomly assigned between 50 and 100.

The traditional methodology is modeled by selecting a random operator among ten operators at each instance, since the TSs do not maintain a reputation system. The proposed approach chooses the operator with the highest reputation score. The simulation results of this experiment are depicted in Fig. 12.

Based on Fig. 12, our system is less prone to errors compared with the TS. This is mainly because we select the

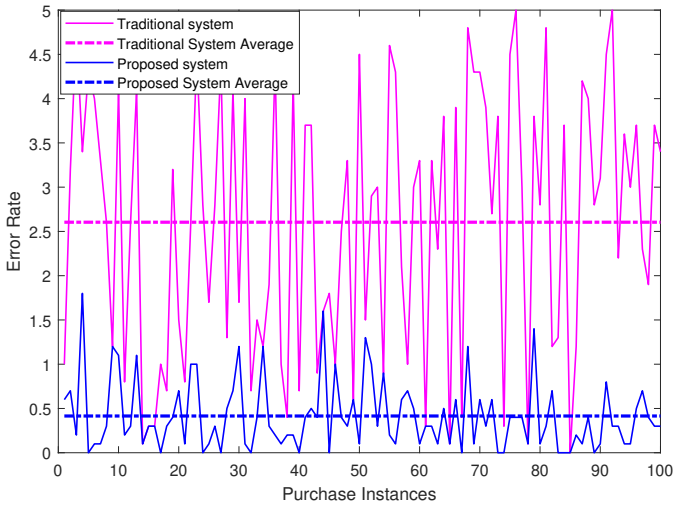


Fig. 12: Error rate vs purchase instances.

seller with the highest reputation score; such operators try to maintain their standard levels while avoiding mistakes.

Subsequently, the average error rates of the current and proposed models are measured by varying the reputation deviation range from 0 to 100 and setting the global error rate to 0.1, 0.05 and 0.01. The tested results are plotted in Fig. 13.

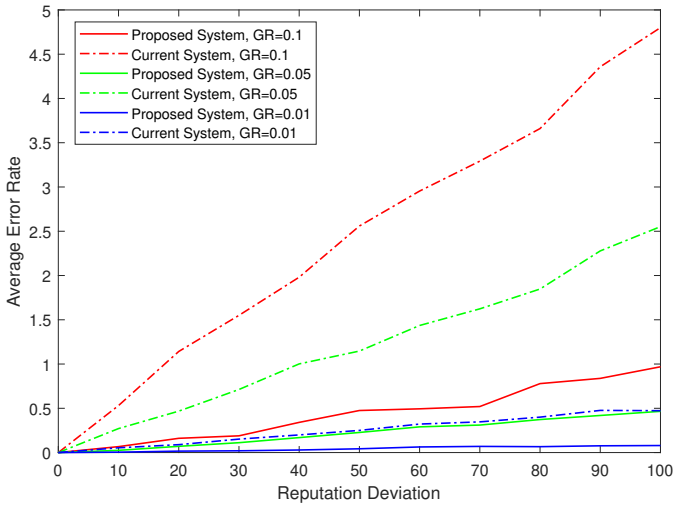


Fig. 13: Average error rate vs reputation deviation.

Based on Fig. 13, a minimal change of average error rate is observed in the proposed system compared with the current model, by varying the percentage of reputation range. Therefore, the reputation variation that exists between VNF sellers does not impact the service quality significantly. In addition, the average error rate of the traditional model rises greatly with a slight increment of global error rate, due to the existence of bottlenecks in the seller selection procedure. Conversely, the proposed model depicts only a modest upsurge by increasing the global error rate moderately. Furthermore, the average error rate is negligible when the global error rate is set to 0.1. Therefore, our system is less vulnerable to errors

and far more beneficial than the current model. The is due to the fact that the seller selection algorithm is based on the reputation.

VIII. Implementation

This section presents the prototypical implementation and the smart contract deployment of the proposed BaaS architecture.

A. Prototype

A prototype of the proposed BaaS architecture has been developed to verify the practical viability. Fig. 14 illustrates the implementation test bed. We performed experimental evaluation in a near realistic environment. The Rinkeby test network was used as the blockchain service hosted in the cloud. The third-party customers were simulated using Raspberry Pi devices over Wi-Fi connectivity to the TCL router. The TCL router connected to the internet using 5G Test Network. The L5GOs are deployed as Virtual Machines (VMs) on Lenovo Thinkpad.

Rinkeby Testnet is an alternative to the main blockchain, which is designed for carrying out experiments [44]. Testnet Ether coins is the form of payment to execute requested operations in the network, which do not have any value. This permits developers to experiment without paying any currency. Currently, there are different types of testnets available and vary only by the employed consensus algorithm. The Rinkeby Testnet utilizes a Proof of Authority (PoA) algorithm. It is controlled by centralized nodes which could be shut down at any time. Thus, it is acceptable for testing purposes only.

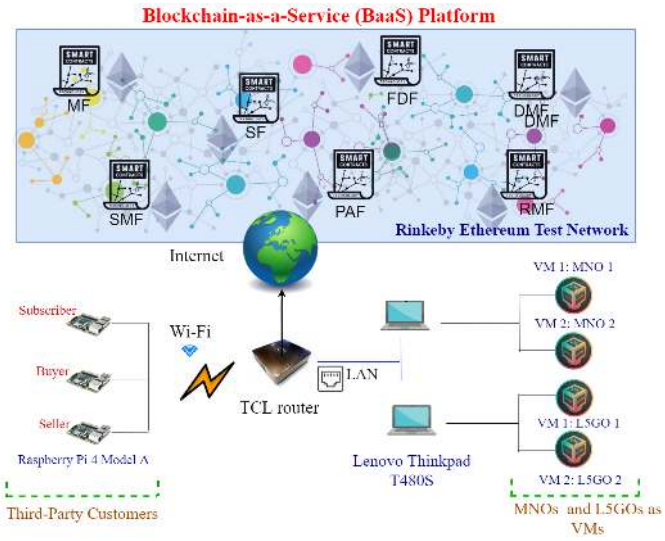


Fig. 14: Implementation Testbed.

Fig. 15 shows two key software elements of the proposed model: the Front-end Client Application and the Decentralized Back-end Server. The front-end client programs were run as HTTP servers that we have deployed in the local host by using the NPM tool. Participants were given access to interact with the blockchain by means of Decentralized Applications (DApps), which are run on a web browser with the MetaMask plugin installed. Metamask acts as a link between the application and the Ethereum blockchain. All the message transfers to and from Ethereum blockchain are performed using the Remote Procedure Call (RPC) protocol. Web3.js is a collection of libraries which makes the communication between DApp and the back-end server possible. Moreover, the front-end application runs on the decentralized back end server, which is the Ethereum blockchain, where all the smart contracts are deployed. Deployed smart contracts manage all the transactions, thereby facilitating all required function calls needed to run roaming, offloading and marketplace functionalities.

The transaction simulation performed using Node JS based javascript programs. The simulation included transactions launched from subscribers, MNOs, L5GO, sellers, and buyers. Metamask communicates with the Ethereum network to perform transactions. The end-to-end latency was measured calculating the difference between transaction initiation and transaction completion.

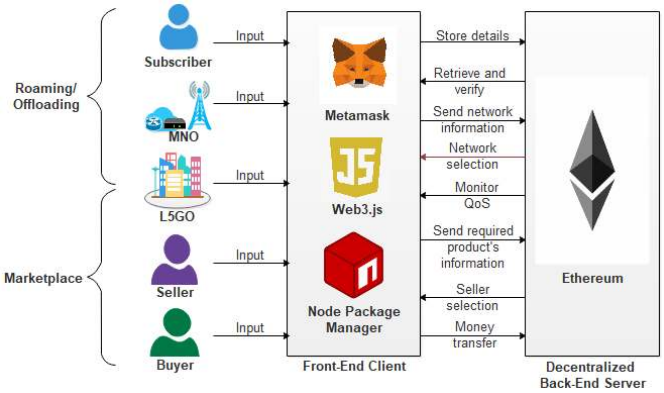


Fig. 15: The Application Architecture of the Prototype.

We ran several tests on this platform to validate the accuracy and to evaluate the performance of the developed DApp.

B. Deployment of Smart Contracts

A prototype of the proposed platform was implemented using Ethereum-based smart contracts. Fig. 16 represents the interaction between these smart contracts. Moreover, the variables and the functions used in each smart contract are detailed in the Appendix. Codes of smart contracts were written in solidity language by using Remix IDE.

1) User Registration Contract: The main purpose of this contract is to register new Tenants while avoiding duplicates. Only MNOs have the permission to register their subscribers to the blockchain. All the user details will be stored in the distributed ledger and shared among the connected blockchain nodes. Therefore, the user details can be retrieved at any given time by sending the IMSI (International Mobile Subscriber Identity) to the blockchain. Furthermore, a user verification function is implemented here. It checks whether the user has already registered in the blockchain network and prevents unauthorized access to the system. The variables and functions used in the user registration contract are listed in Table XVI.

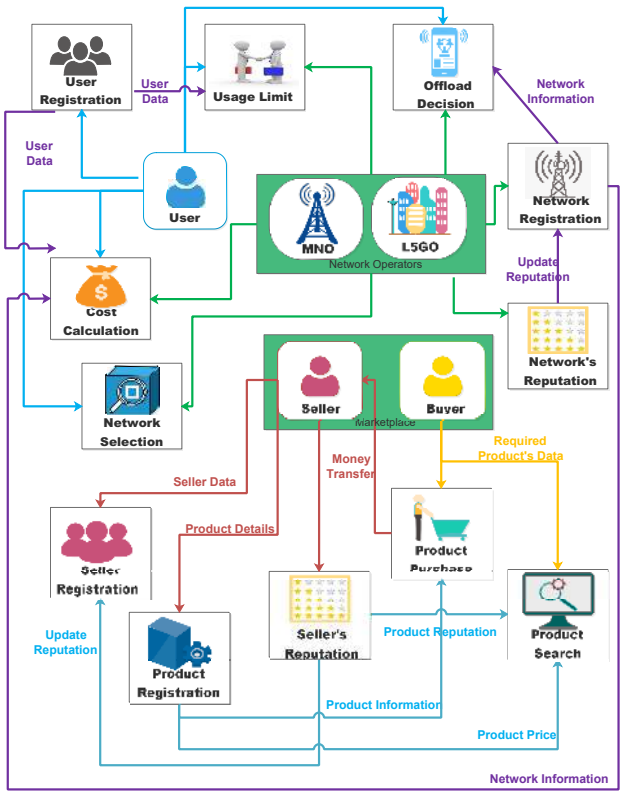


Fig. 16: Interaction between smart contracts.

2) Network Registration Contract: The role of this contract is to register MNOs and L5GOs. For each network, the respective capacity, bandwidth, reputation, and charging schemes are recorded. The structure of this contract is recorded in Table XVII.

3) Offload Decision Contract: This contract is executed to perform the offload process. It calculates the offload scores and returns the L5GO with the highest score. The structure of this contract is recorded in Table XVIII.

4) Network Selection Contract: The main purpose of this contract is to find the best available network for a roaming user. It is initiated when a user starts to send details of all the nearby available networks along with their signal strengths. Furthermore, it calculates roaming scores for all the possible L5GOs. Then, the L5GO with the highest score is returned. The structure of this contract is listed in Table XIX.

5) Network Reputation Management Contract: The contract is invoked whenever a session is ended. The functionality of this contract is to compute a reputation score for each connected network provider and update the score to the blockchain. The structure of this contract is shown in Table XX.

6) Usage Limit Contract: This smart contract acts as the dynamic agreement between the MNO and the L5GO. The L5GO is strictly responsible to deliver the network services based on the agreement. The structure of this contract is recorded in Table XXI.

7) Cost Calculation Contract: The main role of this smart contract is to provide billing information related to user consumption and reputation-based incentives or penalties for L5GOs. Failing to maintain the minimum standard will result in penalties, while exceeding the satisfactory level will be rewarded with incentives. Penalties or incentives will be deducted from or added to the operators' accounts. The structure of this contract is tabulated in Table XXII.

8) Seller Registration Contract: This contract is mainly used to register details of the sellers in the marketplace domain, and it takes measures to prevent seller replication. The initial reputation score is set to the system's average reputation score to ensure fairness among sellers. The structure of this contract is listed in Table XXIII.

9) Product Registration Contract: The sole purpose of this contract is to register and store data of the selling products in the distributed ledger. The product creation is restricted only for the registered sellers. The structure of this contract is recorded in Table XXIV.

10) Search Product Contract: This contract returns the ID of best matching products for each buyer depending upon their requirements. The structure of this contract is tabulated in Table XXV.

11) Product Purchase Contract: The methodology for direct product purchasing is executed in this contract. Mainly, the transfer of the product ownership from a specific seller to another particular buyer is enabled through the Product Purchase contract. The structure of this contract is recorded in Table XXVI.

12) Seller Reputation Management Contract: The main objective of this contract is to compute the reputation score for every seller involved in the marketplace domain. The structure of this contract is recorded in Table XXVII.

IX. Experimental Results

Several tests were conducted on the prototype testbed to analyze the performance of proposed BaaS architecture. The proposed model was executed separately for marketplace and roaming and offloading applications.

A. Roaming and Offloading

The proposed roaming and offloading platform was modeled using a DApp and executed via Ethereum-based smart contracts. The developed platform was evaluated by running 100 tests in Rinkeby Testnet. The performance of the proposed system was measured with regard to latency and cost.

1) Roaming Delay: Initially, codes were written on smart contracts, then deployed to the Ethereum blockchain. For this particular model, smart contracts were written for the network selection and fraud prevention components. End-to-end latency of the roaming process is the summation of the time taken to implement smart contracts corresponding to network selection and fraud prevention components, and the hand-off latency of 50ms [45]. Next, the latency measurements were taken and then plotted in Fig. 17 with a 95% confidence interval.

2) Offload Delay: Offload delay is the summation of the time taken to execute the offload mechanism and the dynamic agreement and the hand-off latency. The existing systems' handover latency for a non-roaming situation is approximately 20ms [45].

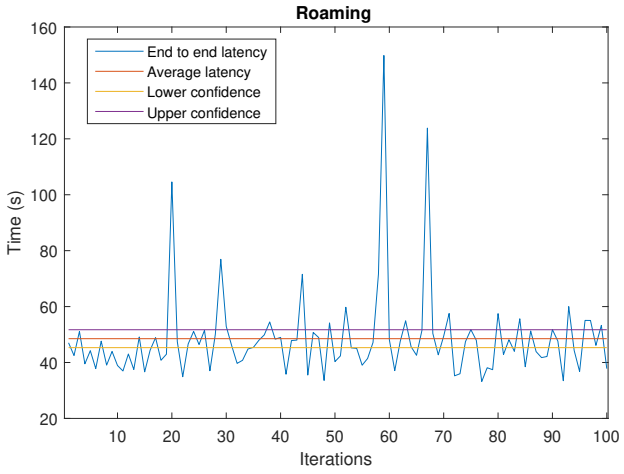


Fig. 17: End to end latency of roaming process.

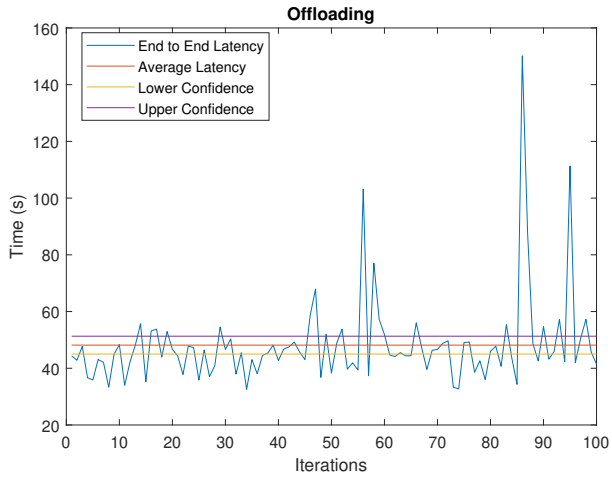


Fig. 18: End to end latency of offloading process.

Based on Fig. 17 and Fig. 18, the average time period to trigger a roaming and offloading instances are 41.3s and 47.7s respectively. However, traditional model shows a roaming delay of approximately 1.75–3.5s [25]. Therefore, it is obvious that our proposed method consists of higher delay than the legacy model. The main factors affecting the roaming delay are the execution of a selection procedure to connect the user for an optimal network and the execution of fraud preventive mechanisms. These processors happen before the migration happens. Thus, this delay is not critical. Moreover, the calculated latency of the proposed model involves the block verification time of 15s [44], which can be further reduced by moving to an optimal consensus algorithm. The appearance of sudden peak levels is due to the latency of the Internet service provider and the processing delay.

3) Cost Analysis: Two types of costs encountered when deploying a smart contract on Ethereum are transaction cost and execution cost. The transaction cost is the gas consumed when a smart contract is sent for validation along with necessary data whereas the execution cost is the gas consumed for executing a smart contract. Costs for each contract are found in the Remix IDE and they are listed in Table XII.

TABLE XII: Cost Evaluation for Roaming and Offloading Applications

Contract Name	**Execution Cost**		**Transaction Cost**	
	Gwei	EUR[a]	Gwei	EUR[a]
User Registration	111415	0,1282	928099	1,068
Network Registration	52050	0,0598	1287451	1,4815
Offload Decision	88373	0,1016	792045	0,9114
Network Selection	68954	0.0793	631856	0,7271
Usage Limit	27782	0,0319	228536	0,2629
Reputation Management	58553	0,0673	466961	0,5373
Cost Calculation	52504	0,0604	474746	0,5463

1 Ether = 10^9 Gwei,[a]1 ether = EUR 1150,80 on 29.01.2021

From the experimental results, the total cost to execute all the proposed functions is less than 2.4 Euro, which is quite low. Therefore, our approach can be considered an economical model. This cost can be further reduced by using a cheaper blockchain platform or creating a permissioned blockchain.

B. Marketplace

For the deployment of the marketplace concept, the following smart contracts were invoked in the Remix IDE: seller registration, product registration, search product, product purchase, and reputation management. To evaluate the proposed method, spectrum sharing, VNF Management, and IoT data sharing applications were considered.The stakeholder inputs were sent to the private blockchain through the DApp and then the corresponding smart contracts for received input were invoked.

The performance of the marketplace framework was tested based on latency and cost. The latency measurements were taken by considering a scenario—that is, with regard to a product querying setting. To find the average time taken to query the list of products, the same experiment with different inputs was run for 100 times in the Rinkeby test network. Such tests were run for spectrum sharing, VNF Management and IoT data sharing applications separately. The results were obtained with a 95% confidence interval. The cost performance was evaluated by listing down the consumed gas for each smart contract execution when deploying marketplace services.

1) Spectrum Sharing: The end-to-end latency to query the selected spectrum within the spectrum-sharing domain is depicted in Fig. 19.

The resulting costs via the execution of the spectrum sharing methodology are listed in Table XIII.

2) VNF Management: Latency measurements obtained by triggering the smart contracts related to VNF Management application are plotted in Fig. 20.

The costs involved in the VNF management operations are recorded in TableXIV.

3) IoT Data Sharing: The time taken to execute IoT data sharing functionalities with regard to querying IoT data from a selling party is estimated and shown in Fig. 21.

The computed costs with the execution of IoT data sharing scheme are given below in Table XV.

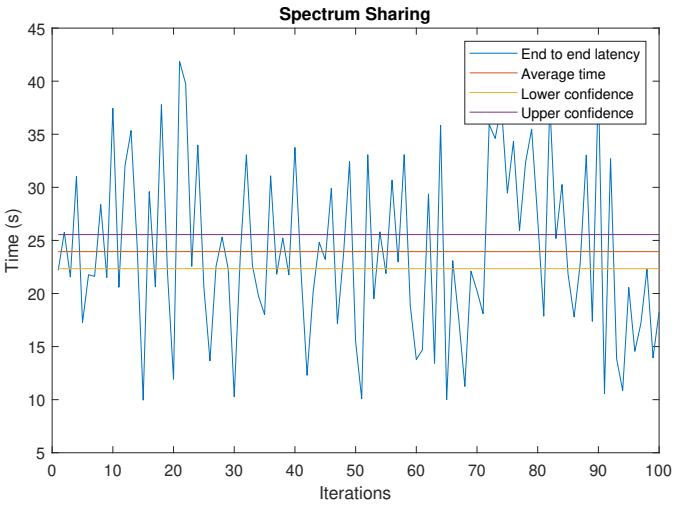


Fig. 19: End to end latency of spectrum sharing.

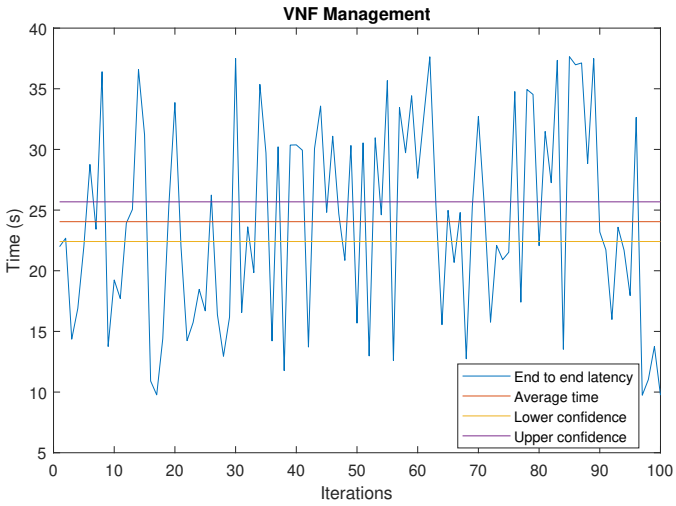


Fig. 20: End to end latency of VNF management.

TABLE XIII: Cost Evaluation for Spectrum Sharing.

Functionalities	**Spectrum Sharing**	
	Gwei	EUR[a]
Seller Registration	84592	0,0973
Product Registration	203243	0,2338
Search Product	28819	0,0331
Product Purchase	34094	0,0392
Reputation Management	31117	0,0358

1 Ether = 10^9 Gwei,[a]1 ether = EUR 1150,80 on 29.01.2021

TABLE XIV: Cost Evaluation for VNF Management.

Functionalities	**VNF Management**	
	Gwei	EUR[a]
Seller Registration	84548	0,0972
Product Registration	314166	0,3615
Search Product	24218	0,0278
Product Purchase	34441	0,0396
Reputation Management	39534	0,0454

1 Ether = 10^9 Gwei,[a]1 ether = EUR 1150,80 on 29.01.2021

Based on Figs. 19, 20, and 21, the average product querying time in spectrum sharing, VNF management, and IoT data sharing are 22.7s, 24.2s and 23.6s, respectively. Therefore, it is apparent that all the applications show almost the same delay, since the same smart contract (Search Product Contract) was invoked. Only the executed internal functions were varied with the application (refer to Table XXVI). Furthermore, 15s out of total time is consumed for block verification. This delay can be further improved by enforcing an optimal consensus algorithm with faster blocktime, or by moving to another blockchain platform like hyperledger, where we can adjust block verification time.

Based on Tables XIII, XIV and XV, the costs incurred to execute marketplace operations are quite low. The total cost to execute one application with all the operations in the marketplace domain is less than 1 Euro (summation of gas consumption to execute each smart contract). Therefore, this model can be considered a cost-beneficial model. However, this cost can also be further reduced by using a cheaper blockchain platform or creating permissioned blockchain.

X. Conclusion

L5GOs are one of the most powerful 5G techniques, with distinguishing potential in different application contexts. We identified the blockchain as one of the most promising technological enablers to cater to future telecommunication demands. Blockchain, with its key enabling features, can be used to fulfill the requirements of an L5GO ecosystem, as we explained comprehensively. Potential blockchain-based opportunities for L5GOs are explored. Challenges in each opportunity are outlined and solutions are suggested to overcome them. A BaaS architecture is proposed by combining all proposals. The proposed approach is evaluated on a Matlab simulation tool and Rinkeby Testnet. Through the simulation results, it is evident that our model is cost effective, with improved QoS compared with the existing roaming system. Furthermore, the deployed reputation management system with regard to Marketplace shows a positive impact on the selection procedure of a seller, which again proves the importance of our model. To measure the functional performance of the proposed system, a DApp was built with the help of the web3.js library. Upon comparison of the obtained latency and cost measurements with the state of art, our model yields a lower latency and is beneficial from the cost perspective.

Acknowledgment

This work is partly supported by European Union in RESPONSE 5G (Grant No: 789658), Academy of Finland in 6Genesis (Grant no. 318927) and 5GEAR (Grant no. 319669) projects.

References

[1] C. Forecast, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper," *Cisco Public Information*, 2017.

[2] M. Agiwal, A. Roy, and N. Saxena, "Next Generation 5G Wireless Networks: a Comprehensive Survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1617–1655, 2016.

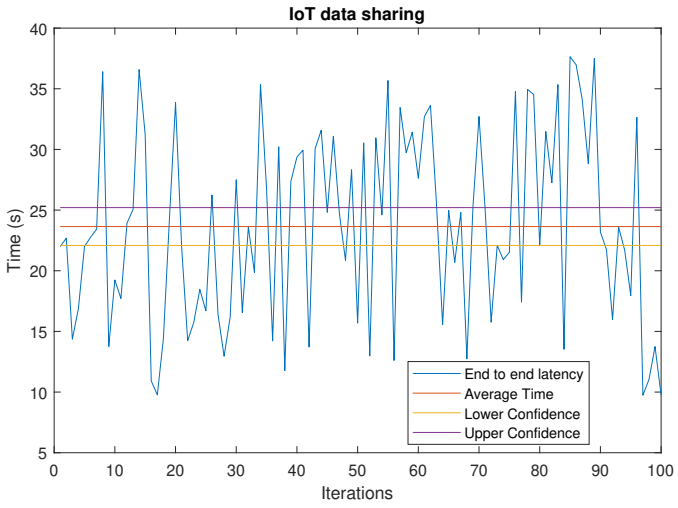


Fig. 21: End to end latency of IoT data sharing.

TABLE XV: Cost Evaluation for IoT Data Sharing

Functionalities	**IoT Data Sharing**	
	Gwei	EUR[a]
Seller Registration	84592	0,0973
Product Registration	241206	0,2775
Search Product	22682	0,0261
Product Purchase	36725	0,0422
Reputation Management	31117	0,0358

1 Ether = 10^9 Gwei,[a]1 ether = EUR 1150,80 on 29.01.2021

[3] P. Marsch, I. Da Silva, O. Bulakci, M. Tesanovic, S. E. El Ayoubi, T. Rosowski, A. Kaloxylos, and M. Boldi, "5G Radio Access Network Architecture: Design Guidelines and Key Considerations," *IEEE Communications Magazine*, vol. 54, no. 11, pp. 24–32, 2016.

[4] A. Prasad, Z. Li, S. Holtmanns, and M. A. Uusitalo, "5G Micro-Operator Networks—A Key Enabler for New Verticals and Markets," in *2017 25th Telecommunication Forum (TELFOR)*. IEEE, 2017, pp. 1–4.

[5] M. Matinmikko-Blue and M. Latva-aho, "Micro Operators Accelerating 5G Deployment," in *2017 IEEE International Conference on Industrial and Information Systems (ICIIS)*. IEEE, 2017, pp. 1–5.

[6] H. Wang, K. Chen, and D. Xu, "A maturity model for blockchain adoption," *Financial Innovation*, vol. 2, no. 1, p. 12, 2016.

[7] H. Natarajan, S. Krause, and H. Gradstein, *Distributed ledger technology and blockchain*. World Bank, 2017.

[8] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE international congress on big data (BigData congress)*. IEEE, 2017, pp. 557–564.

[9] Z. Tian, M. Li, M. Qiu, Y. Sun, and S. Su, "Block-DEF: A secure digital evidence framework using blockchain," *Information Sciences*, vol. 491, pp. 151–165, 2019.

[10] L. W. Cong and Z. He, "Blockchain disruption and smart contracts," *The Review of Financial Studies*, vol. 32, no. 5, pp. 1754–1797, 2019.

[11] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in *2016 IEEE symposium on security and privacy (SP)*. IEEE, 2016, pp. 839–858.

[12] G. Praveen, V. Chamola, V. Hassija, and N. Kumar, "Blockchain for 5G: A Prelude to Future Telecommunication," *IEEE Network*, 2020.

[13] V. K. Rathi, V. Chaudhary, N. K. Rajput, B. Ahuja, A. K. Jaiswal, D. Gupta, M. Elhoseny, and M. Hammoudeh, "A Blockchain-Enabled Multi Domain Edge Computing Orchestrator," *IEEE Internet of Things Magazine*, vol. 3, no. 2, pp. 30–36, 2020.

[14] M. Saravanan, S. Behera, and V. Iyer, "Smart Contracts in Mobile Telecom Networks," in *2017 23RD Annual International Conference in Advanced Computing and Communications (ADCOM)*. IEEE, 2017, pp. 27–33.

[15] C. T. Nguyen, D. N. Nguyen, D. T. Hoang, H.-A. Pham, N. H. Tuong, and E. Dutkiewicz, "Blockchain and stackelberg game model for roaming fraud prevention and profit maximization," in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2020, pp. 1–6.

[16] N. Weerasinghe, T. Hewa, M. Dissanayake, M. Ylianttila, and M. Liyanage, "Blockchain-based Roaming and Offload Service Platform for Local 5G Operators."

[17] M. Matinmikko-Blue, S. Yrjölä, V. Seppänen, P. Ahokangas, H. Hämmäinen, and M. Latva-aho, "Analysis of Spectrum Valuation Approaches: The Viewpoint of Local 5G Networks in Shared Spectrum Bands," in *2018 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2018, pp. 1–9.

[18] D. He, C. Chen, J. Bu, S. Chan, and Y. Zhang, "Security and Efficiency in Roaming Services for Wireless Networks: Challenges, Approaches, and Prospects," *IEEE Communications Magazine*, vol. 51, no. 2, pp. 142–150, 2013.

[19] G. Macia-Fernandez, P. Garcia-Teodoro, and J. Diaz-Verdejo, "Fraud in Roaming Scenarios: an Overview," *IEEE Wireless Communications*, vol. 16, no. 6, pp. 88–94, 2009.

[20] G. Liu and H. Zhao, "Power Allocation and Channel Selection in Small Cell Networks Based on Traffic-Offloading," in *2017 First International Conference on Electronics Instrumentation & Information Systems (EIIS)*. IEEE, 2017, pp. 1–4.

[21] M. Matinmikko, M. Latva-Aho, P. Ahokangas, S. Yrjölä, and T. Koivumäki, "Micro Operators to Boost Local Service Delivery in 5G," *Wireless Personal Communications*, vol. 95, no. 1, pp. 69–82, 2017.

[22] S. Han and X. Zhu, "Blockchain Based Spectrum Sharing Algorithm," in *2019 IEEE 19th International Conference on Communication Technology (ICCT)*. IEEE, 2019, pp. 936–940.

[23] M. B. Weiss, K. Werbach, D. C. Sicker, and C. E. C. Bastidas, "On the application of blockchains to spectrum management," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 2, pp. 193–205, 2019.

[24] S. Zheng, T. Han, Y. Jiang, and X. Ge, "Smart Contract-Based Spectrum Sharing Transactions for Multi-Operators Wireless Communication Networks," *IEEE Access*, vol. 8, pp. 88 547–88 557, 2020.

[25] A. Refaey, K. Hammad, S. Magierowski, and E. Hossain, "A Blockchain Policy and Charging Control Framework for Roaming in Cellular Networks," *IEEE Network*, 2019.

[26] S. Raju, S. Boddepalli, S. Gampa, Q. Yan, and J. S. Deogun, "Identity management using blockchain for cognitive cellular networks," in *2017 IEEE International Conference on Communications (ICC)*. IEEE, 2017, pp. 1–6.

[27] Z. Haddad, M. M. Fouda, M. Mahmoud, and M. Abdallah, "Blockchain-based Authentication for 5G Networks," in *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*. IEEE, 2020, pp. 189–194.

[28] M. F. Franco, E. J. Scheid, L. Z. Granville, and B. Stiller, "BRAIN: blockchain-based reverse auction for infrastructure supply in virtual network functions-as-a-service," in *2019 IFIP Networking Conference (IFIP Networking)*. IEEE, 2019, pp. 1–9.

[29] G. A. F. Rebello, G. F. Camilo, L. G. Silva, L. C. Guimarães, L. A. C. de Souza, I. D. Alvarenga, and O. C. M. Duarte, "Providing a sliced, secure, and isolated software infrastructure of virtual functions through blockchain technology," in *2019 IEEE 20th International Conference on High Performance Switching and Routing (HPSR)*. IEEE, 2019, pp. 1–6.

[30] Y.-W. Chang, K.-P. Lin, and C.-Y. Shen, "Blockchain Technology for e-Marketplace," in *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 2019, pp. 429–430.

[31] H. Desai, M. Kantarcioglu, and L. Kagal, "A Hybrid Blockchain Architecture for Privacy-Enabled and Accountable Auctions," in *2019 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 2019, pp. 34–43.

[32] J. Thakker, I. Chang, and Y. Park, "Secure Data Management in Internet-of-Things Based on Blockchain," in *2020 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 2020, pp. 1–5.

[33] G. Ayoade, V. Karande, L. Khan, and K. Hamlen, "Decentralized IoT data management using blockchain and trusted execution environment," in *2018 IEEE International Conference on Information Reuse and Integration (IRI)*. IEEE, 2018, pp. 15–22.

[34] H. Jeon and B. Lee, "Network Service Chaining Challenges for VNF Outsourcing in Network Function Virtualization," in *2015 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2015, pp. 819–821.

[35] K. Shafique, B. A. Khawaja, F. Sabir, S. Qazi, and M. Mustaqim, "Internet of Things (IoT) for Next-Generation Smart Systems: a Review of Current Challenges, Future Trends and Prospects for Emerging 5G-IoT Scenarios," *IEEE Access*, vol. 8, pp. 23 022–23 040, 2020.
[36] "MATLAB," last accessed 23 May 2020. [Online]. Available: https://www.mathworks.com
[37] M. Falch, A. Henten, and R. Tadayoni, "International roaming: is there a need for EU-regulation beyond 2010?" *info*, 2009.
[38] "Roaming Rates," last accessed 29 October 2020. [Online]. Available: https://europa.eu/youreurope/citizens/consumers/internet-telecoms/mobile-roaming-costs/index_en.htm
[39] S. Forge and L. Srivastava, "ITU cost model and methodology to assist national regulatory authorities to engage with international mobile roaming," *Digital Policy, Regulation and Governance*, 2018.
[40] V. Ntarzanou and M. Portela, "Telecom operators and the aftermath of the European Commission agenda for the termination of roaming charges within the EU," 2015.
[41] "International Mobile Roaming Strategic Guidelines 2017," last accessed 28 October 2020. [Online]. Available: https://www.itu.int/en/ITU-D/Conferences/GSR/Documents/GSR2017/IMR_Strategic_Guidelines_Second%20Consultation_DRAFT_FINAL.pdf
[42] D. Lloyd, "International Roaming Fraud: Trends and Prevention Techniques," *Fair Isaac Corporation*, 2003.
[43] "Tax Rates," last accessed 29 October 2020. [Online]. Available: https://www.dialog.lk/tax
[44] "Ethereum Testnet," last accessed 14 May 2020. [Online]. Available: https://www.rinkeby.io/
[45] C. Cox, *an Introduction to LTE: LTE, LTE-Advanced, SAE and 4G Mobile Communications*. John Wiley & Sons, 2012.

Nisita Weerasinghe is currently working as a Doctoral Student at the Centre for Wireless Communications (CWC), University of Oulu, Finland. She received her B.Sc degree in Electrical and Electronic Engineering from Sri Lanka Institute of Information Technology (SLIIT), Malabe, Sri Lanka, in 2018 and Master of Science in Wireless Communications Engineering from the University of Oulu, Finland, in 2020. Her research interests include Blockchains, IoT and network security, mobile communication systems and data communications.

Tharaka Hewa is currently working as a Doctoral Student in Centre for Wireless Communication, University of Oulu, Finland. He received his Bachelor's degree in Computer Science from the University of Colombo School of Computing, Sri Lanka in 2013, and Master of Science in Information Security (Distinction) from the University of Colombo School of Computing in 2016. From 2012 to 2017, he worked in a leading digital payment solution provider in Sri Lanka as a Senior Software Engineer. Within his career in the industry, he contributed to many projects in mobile banking, internet banking, PKI, Automated Teller Machines and involved in the system integration and support. He is a certified engineer for SafeNet Luna SA 6.0 HSM. In 2017, he joined Nanyang Technological University as a Research Associate. He played a vital role in many research and implementation projects in different contexts. He contributed to cybersecurity and digital payment systems and co-authored 2 publications related to the blockchain applications in the industry with contributing to 1 patent. He contributed to ongoing research and implementation projects including blockchain for 3D printing, agriculture, luxury watches, music asset monetization, and aviation. In 2019 he joined Centre for Wireless Communication and his research focus in developing a blockchain-based platform as a service for industrial IoT.

Hewa's research interests are Blockchain, PKI, 5G, Banking Systems Security, Healthcare Security, and Smart Cities.

Madhusanka Liyanage (Senior Member, IEEE) received his B.Sc. degree (First Class Honours) in electronics and telecommunication engineering from the University of Moratuwa, Moratuwa, Sri Lanka, in 2009, the M.Eng. degree from the Asian Institute of Technology, Bangkok, Thailand, in 2011, the M.Sc. degree from the University of Nice Sophia Antipolis, Nice, France, in 2011, and the Doctor of Technology degree in communication engineering from the University of Oulu, Oulu, Finland, in 2016. From 2011 to 2012, he worked a Research Scientist at the I3S Laboratory and Inria, Shopia Antipolis, France. He is currently an assistant professor/Ad Astra Fellow at School of Computer Science, University College Dublin, Ireland. He is also acting as an adjunct Processor at the Center for Wireless Communications, University of Oulu, Finland. He was also a recipient of prestigious Marie Skłodowska-Curie Actions Individual Fellowship during 2018-2020. During 2015-2018, he has been a Visiting Research Fellow at the CSIRO, Australia, the Infolabs21, Lancaster University, U.K., Computer Science and Engineering, The University of New South Wales, Australia, School of IT, University of Sydney, Australia, LIP6, Sorbonne University, France and Computer Science and Engineering, The University of Oxford, U.K. He is also a senior member of IEEE. In 2020, he has received "2020 IEEE ComSoc Outstanding Young Researcher" award by IEEE ComSoc EMEA. Dr. Liyanage's research interests are 5G/6G, SDN, IoT, Blockchain, MEC, mobile and virtual network security. More info: www.madhusanka.com

Salil S. Kanhere (Senior Member, IEEE) received his M.S. and Ph.D. degrees, both in Electrical Engineering from Drexel University, Philadelphia. He is a Professor in the School of Computer Science and Engineering at UNSW Sydney, Australia. He also holds affiliations with CSIRO's Data61 and Cybersecurity Cooperative Research Centre. He has held visiting appointments with Institute of Infocom Research Singapore, Technical University Darmstadt, University of Zurich and Graz University of Technology. His research interests include Internet of Things, pervasive computing, cyberphysical systems, blockchain, cybersecurity and applied machine learning. He has published over 250 peer-reviewed articles and delivered over 50 tutorials and keynote talks on these research topics. He has received 8 Best Paper Awards. His research has been featured on ABC News Australia, Forbes, Wired, ZDNET, MIT Technology Review, Computer World, IEEE Spectrum and other media outlets. He regularly features on the organizing committee of a number of IEEE and ACM international conferences. He is the General Chair for the IEEE International Conference on Blockchain and Cryptocurrency (IEEE ICBC) in 2021. Salil is the Editor in Chief of the Ad Hoc Networks Journal and on the Editorial Board of IEEE Transactions on Network Management and Service, Pervasive and Mobile Computing and Computer Communications. He serves on the Executive Committee of the IEEE Computer Society's Technical Committee on Computer Communications (TCCC). Salil is a Senior Member of both the IEEE and the ACM. He is a recipient of the Alexander von Humboldt Research Fellowship.

Mika Ylianttila (Senior Member, IEEE) is a full-time associate professor (tenure track) at the Centre for Wireless Communications (CWC), at the Faculty of Information Technology and Electrical Engineering (ITEE), University of Oulu, Finland. He is leading a research team and is the director of communications engineering doctoral degree program. Previously he was the director of Center for Internet Excellence (2012-2015), vice director of MediaTeam Oulu research group (2009-2011), and professor (pro tem) in computer science and engineering, and director of information networks study programme (2005-2010). He received his doctoral degree on Communications Engineering at the University of Oulu in 2005. He has coauthored more than 150 international peer-reviewed articles. His research interests include edge computing, network security, network virtualization and software-defined networking. He is a Senior Member of IEEE, and Editor in *Wireless Networks* journal.

APPENDIX
THE STRUCTURE OF DEPLOYED SMART CONTRACTS

TABLE XVI: The structure of the User Registration contract

Variables		
Type	Name	Description
struct	User	To store the details of the subscriber.
address	Owner	To store the address of the current network provider who is accessing the system.
address[]	addressTousers	Public array comprising subscriber information mapped to their Ethereum addresses.
Functions		
Name	Description	
onlyOwner	Modifier that allows only a network operator to add the details of a user.	
notRegistered	Modifier that checks whether the user has registered already.	
createUser	If the conditions written in the ”onlyOwner” and ”notRegistered” modifier functions satisfy, this function will register user details, the values of IMSI, name and home address to the user structure	
verifyUser	Function checks the validity of the on-board subscribers using their IMSI number.	
getUserBalance	Returns the available balance in the user’s universal wallet, attached to their IMSI number.	
deductUserBalance	Total charge of consumption is deducted from the subscriber’s remaining account balance.	

TABLE XVII: The structure of the Network Registration contract

Variables		
Type	Name	Description
struct	Network	To store details of the network
address	Owner	To store the address of the current network provider who is accessing the system
address[]	addressTonetworks	Public array comprising network information mapped to their Ethereum addresses
int256	averageReputation	Global variable to store the average reputation of the prevailing system
Functions		
Name	Description	
onlyOwner	Modifier that allows only a network operator to add the details of a user	
notRegistered	Modifier that checks whether the user has registered already	
registerNetwork	If the conditions given in the ”onlyOwner” and ”notRegistered” modifier functions satisfy, this function will register the network details	
getReputation	Returns the reputation of a network provider given its Ethereum address	
getCostRating	Returns the cost rating factor of a network provider given its Ethereum address	
getNetworkCapacity	Returns the capacity of a network provider given its Ethereum address	
getNetworkBandwidth	Returns the bandwidth of a network provider given its Ethereum address	
getNetworkName	Returns the registered name of a network provider given its Ethereum address	
getNetworkCount	Returns the total number of registered network providers	
getCostWeight	Returns the predefined weight of the cost parameter	
getReputationWeight	Returns the predefined weight of the reputation parameter	
getCapacityWeight	Returns the predefined weight of the network capacity parameter	
getBandwidthWeight	Returns the predefined weight of the network bandwidth parameter	
getStrengthWeight	Returns the predefined weight of the signal strength parameter	
getMNOCallCost	Returns the predefined call cost of a given network	
getMNOSmsCost	Returns the predefined sms cost of a given network	
getMNODataCost	Returns the predefined data cost of a given network	
updateAvgReputation	Update the average reputation of the system at the end of every session	
updateReputation	Update the reputation score of a network given its Ethereum address	

TABLE XVIII: The structure of the Offload Decision contract

Variables		
Type	Name	Description
NetworkRegisterContract	networkcontract	Instance of the deployed Network Registration contract.
Functions		
Name	Description	
offloadDecision	Creates a instance of the network registration contract using the deployed ”networkcontract” address. Perform the functionalities related to offload service as described in the section V-4	
selectedNetwork	Event function that returns the name and the offload score of the optimum network provider	

TABLE XIX: The structure of the Network Selection contract

Variables		
Type	Name	Description
NetworkRegisterContract	networkcontract	Instance of the deployed Network Registration contract
struct[]	DetectedNetworks	To store the network address and the signal strength of the detected nearby networks.
Functions		
Name	Description	
roamingDecision	Creates a instance of the network registration contract using the deployed ”networkcontract” address. Perform the functionalities related to roaming service as described in the section V-4.	
selectedNetwork	Event function that returns the name and the roaming score of the optimum network provider.	

TABLE XX: The structure of the Network’s Reputation Management contract

Variables		
Type	Name	Description
NetworkRegisterContract	networkcontract	Instance of the deployed Network Registration contract
int256	allowedLatency	To store the predefined threshold value of the allowable latency
int256	allowedPL	To store the predefined threshold value of the allowable packet loss
int256	allowedJitter	To store the predefined threshold value of the allowable jitter
int256	allowedBP	To store the predefined threshold value of the blocking probability
Functions		
Name	Description	
reputationManagement	Creates a instance of the network register contract using the deployed ”networkcontract” address. Calculates reputation of a network provider, given its Ethereum address, using predefined performance indexes	
reputationScore	Event function that returns the computed reputation score of a network provider at the end of each session	

TABLE XXI: The structure of the Usage Limit contract

Variables		
Type	Name	Description
RegisterUsersContract	usercontract	Instance of the deployed User Registration contract
Functions		
Name	Description	
getUsageLimit	Creates a instance of the user registration contract using the deployed ”usercontract” address. It returns the maximum limit that a network provider must provide the service to a given user	
usageLimit	Event function that emits the return value of the ”getUsageLimit” function	

TABLE XXII: The structure of the Cost Calculation contract

Variables		
Type	Name	Description
NetworkRegisterContract	networkcontract	Instance of the deployed Network Registration contract
RegisterUsersContract	usercontract	Instance of the deployed User Registration contract
Functions		
Name	Description	
sessionData	Creates instances of network registration contract and user registration contract using the deployed ”networkcontract” address and ”usercontract” address respectively. It calculates the service cost using session data and update the user’s account balance accordingly	
incentivePenalty	Event function that emits the incentive or penalty value for a network provider based on session data	

TABLE XXIII: The structure of the Seller Registration contract

Variables		
Type	Name	Description
address[]	sellerData	Maps the variable to store seller data of type ”struct Sellers”, which links to the seller address as the key data
uint	sellerCount	Keeps track of the registered seller count
Functions		
Name	Description	
registerSeller	Creates a new seller considering calling address as the seller ID and assign an average reputation value	

TABLE XXIV: The structure of the Product Registration contract

Variables		
Type	Name	Description
SellerRegisterContract	sellerContract	Instance of deployed seller registration contract
uint[]	products	Maps variable to store product data of type ”struct Product”, which links to an index value as the key data
uint	productCount	Keeps track of the registered product count
Functions		
Name	Description	
createProduct	Creates a new product with given attributes and save it in ”products” mapping. It also calls the registerSeller function of ”SellerRegisterContract” to save the caller as a new seller	

TABLE XXV: The structure of the Search Product contract

Variables		
Type	Name	Description
SellerRegisterContract	sellerContract	Instance of deployed seller registration contract
ProductRegisterContract	productContract	Instance of deployed product registration contract
Functions		
Name	Description	
createProduct	Creates a new product with given attributes and save it in ”products” mapping. It also calls the registerSeller function of ”SellerRegisterContract” to save the caller as a new seller	
searchIOTProduct	Search database for IOT products of desired category and returns the ID, Seller and price of the highest rated product	
searchVNFProduct	Search database for VNF products of desired VNF Score and returns the ID of the selected product	
searchSSProduct	Search database for Spectrum Sharing products of a desired band and returns the ID of the selected product	
getProductCategory	Returns the productCategory value of a registered product	
getProductID	Returns the productID value of a registered product	
getProductPrice	Returns the productPrice value of a registered product	
getProductVNFScore	Returns the productVNFScore value of a registered product	
getProductOwner	Returns the productOwner value of a registered product	
getProductBandNumber	Returns the productCategory value of a registered product	
getSellerReputation	Returns the reputation value of a registered seller	

TABLE XXVI: The structure of the Product Purchase contract

Variables		
Type	Name	Description
SellerRegisterContract	sellerContract	Instance of deployed seller registration contract
ProductRegisterContract	productContract	Instance of deployed product registration contract
Product	productVar	Variable to save a duplicate of a selected product
address	sellerAddress	Variable to hold an address of a selected product owner
Functions		
Name	Description	
purchaseProduct	Called when a product is purchased. Ownership of the relevant product of the given ID is transferred to the buyer upon calling this function	
fetchProduct	Creates a copy of a given product	
updateProduct	Updates a product in the database with given attributes	

TABLE XXVII: The structure of the Seller's Reputation Management contract

Variables		
Type	Name	Description
SellerRegisterContract	sellerContract	Instance of deployed seller registration contract
ProductRegisterContract	productContract	Instance of deployed product registration contract
uint	reputationIOT	To store the calculated reputation score of an IOT seller
uint	reputationVNF	To store the calculated reputation score of a VNF seller
uint	reputationSS	To store the calculated reputation score of a Spectrum Sharing seller
Functions		
Name	Description	
calculateIOTReputation	Calculates reputation of an IOT seller using predefined performance indexes	
calculateVNFReputation	Calculates reputation of a VNF seller using predefined performance indexes	
calculateSSReputation	Calculates reputation of a Spectrum Sharing seller using predefined performance indexes	
getProductAvailability	Returns the availability value of a registered product	
getProductServiceDuration	Returns the serviceDuration value of a registered product	
getProductChannelQuality	Returns the channelQuality value of a registered product	
getProductMemory	Returns the memory value of a registered product	
getProductDisk	Returns the disk value of a registered product	
getProductCPUCores	Returns the CPUCores value of a registered product	
setSellerReputation	Sets the reputation value of a registered seller	