

Received March 15, 2022, accepted March 29, 2022, date of publication April 5, 2022, date of current version April 13, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3165087

# A Smart Contract for Coffee Transport and Storage With Data Validation

CRISTIAN VALENCIA-PAYAN<sup>1</sup>, JOSÉ FERNANDO GRASS-RAMÍREZ<sup>2</sup>,  
GUSTAVO RAMIREZ-GONZALEZ<sup>1</sup>, AND JUAN CARLOS CORRALES<sup>1</sup>

<sup>1</sup>Department of Telematics, University of Cauca, Cauca, Popayán 190003, Colombia

<sup>2</sup>Department of Agroindustry, University of Cauca, Cauca, Popayán 190003, Colombia

Corresponding author: Cristian Valencia-Payan (chpayan@unicauca.edu.co)

This work was supported in part by the Sistema General de Regalías (SGR) through the Project “Incremento de la Oferta de Prototipos Tecnológicos en Estado Pre-Comercial Derivados de Resultados de I+D Para el Fortalecimiento del Sector Agropecuario en el Departamento del Cauca” (BPIN 202000100098); and in part by the Project “Estrategias Para la Valorización de Los Dulces Tradicionales de Popayán” (Code: 110380863995) founded by Minciencias, Colombia, under Contrato 127-2019.

**ABSTRACT** Recently there has been an increase in the use of Blockchain technology for multiple purposes; one of them has been food traceability. This technology has increased quality control, safety, and reliability. So, the producers are looking for better ways to trace the products at any supply chain stage to ensure their quality. A smart contract is a transaction protocol that execute automatically when a predefined set of conditions are met. In this paper, we propose a smart contract to monitor the status of the coffee beans in the transport and storage stages with data validation. Using the Hyperledger Fabric Blockchain tool, we deploy a test network of two actors, also known as organizations. The organizations come together to form a channel in the network. Each has a valid identity that helps them verify their signatures over any transaction. We selected JavaScript to write our proposed smart contract for experimental and evaluation purposes. To evaluate the smart Contract, we use Hyperledger Caliper, obtaining an average throughput of 10.4tps and average latency of 0.7s, being fast enough to be used in a real environment, considering the current control conditions of the coffee beans.

**INDEX TERMS** Blockchain, coffee, coffee certifications, reliability, smart contract, storage, traceability, transport.

## I. INTRODUCTION

One of the most consumed beverages in the world, after water, is coffee. Millions of people drink coffee daily, and the demand is increasing not only for coffee but for high-quality specialty coffee [1]. Also, coffee is one the most valuable agricultural product in the world, according to the International Coffee Organization (ICO) [2], both in terms of volume and value [3]. Multiple third-party certification schemes have recently been adapted globally like Certified Organic, Rainforest Alliance, among others [4]. Using these certifications, the coffee growers and coffee organizations can sell their product to national and international markets with an increased profit margin. In this context, some authors consider no clear benefit for the smallholders [5] but considers that product traceability and the possibility of direct sale mechanisms that guarantee the origin and preservation of the

quality of the coffee in the value chain could provide higher remuneration to the farmers.

In Colombia, the National Federation of Coffee Growers (FNC, by its acronym in Spanish) [6] has regulated the quality conditions for coffee export and the purchase price penalties for coffee bags based on their quality [7], [8]. Maintaining quality conditions in each coffee bag is necessary to avoid fungi and defects during the transport and storage. Thus, variables like bean moisture on dried parchment coffee, temperature, water activity on dried parchment coffee must be controlled and checked to ensure the quality of the coffee beans.

As mentioned earlier, the selling price of a coffee bag will fluctuate according to the quality conditions of the coffee beans; but also, circumstances such as the uneven distribution of value and risks, increasing labor costs, climate changes, and diseases, has a considerable impact on the livelihood of coffee growers [9].

Recently there has been an increase in the use of Blockchain technology for multiple purposes; one of them

The associate editor coordinating the review of this manuscript and approving it for publication was Cristian Zambelli<sup>1</sup>.

has been the food traceability [10]. Blockchain has been used to improve access to traceable information by making it available for all parties at any time. This technology has increased the agricultural value chain's quality control, safety, and reliability [11]. So, the producers are looking for better ways to trace the products at any chain stage to ensure their quality.

A vital feature of the Blockchain network are smart contracts; once deployed, these contracts run autonomously when the agreed conditions have been met or when the network receives a transaction related to the particular contract deployed [12]. Smart contracts as the Blockchain network are immutable and can be reviewed and updated at any time.

As we mentioned above, coffee beans can lose their quality if transported or stored in unsuitable conditions, impacting the coffee bag price; so, in this paper, we propose a smart contract to monitor the status of the coffee beans during transport and storage. In this way, the farmer can make the right decisions regarding handling before the sale and obtain better profits, considering coffee certification schemes.

This paper is organized as follows: Section 2 introduces key concepts to understand the investigative environment, section 3 shows how Blockchain technology would apply to the traceability of agri-food processes, section 4 presents the smart contract proposed for the coffee chain, specifically in the stages of Transport and storage. Subsequently, section 5 shows the implementation and tests of the smart contract, followed by the conclusions in section 6.

## II. KEY CONCEPTS

### A. SMART CONTRACT

The smart contract was proposed first by Nick Szabo [13]. However, its use began with the introduction of Blockchain technology in 2008. A smart contract is software that runs autonomously once deployed on the Blockchain network [12] when a series of conditions agreed by the members of the network are met.

Organizations like Ethereum have proposed standards for the packaging of smart contracts to facilitate the reuse of developed contracts and have also made efforts to offer standards throughout the construction of the Blockchain network [14].

A smart contract is a computer program that considers all the conditions agreed by the parties. When it is deployed on the Blockchain network, it can be called by any user on the network with access permissions and changes a set of state variables on the network using the transaction information. According to definition 12 on [15], the smart contract needs to satisfy the following equation.

$$C(s_i, Tx_i) = (s_j, R_i) \quad (1)$$

where  $S = \{S_{i \in \mathbb{N}^*}\}$  represents all the possible states in the smart contract  $C$ .  $T = \{Tx_{i \in \mathbb{N}^*} = (t, in, out, s, pld)_{i \in \mathbb{N}^*}\}$  exposes a transaction on the Blockchain network where  $t$  is the transaction's timestamp,  $in$  represent the data sent,  $out$  represent the transaction response,  $s$  represents the

transaction signature and  $pld$  is the payload data.  $R = \{R_{i \in \mathbb{N}^*}\}$  represent the possible responses that the smart contract will give. If a valid transaction calls the contract, the smart contract will produce a new state  $S_j$  and a corresponding response.

### B. TRACEABILITY

Traceability of a product is the ability to identify a product at any stage of the supply chain [16], it can be a "One-up One-Down" implementation, or it may be a little more complex depending on the level of information of the product [17]. It can also be defined as Traceability, the ability to identify and track the history, distribution, location, application of products, parts, and materials of a final product to ensure reliability.

### C. DATA VALIDATION

Data validation is a process to accept or refuse data values in a finite or infinite set of acceptable values [18]. Its goal is to ensure that the data are correct and also helpful. It depends on the validation rules set for the phenomenon represented by the data, although some rules can be used generally.

### D. COFFEE CERTIFICATIONS

Voluntary Sustainability Standards (VSS) are standards developed at the local, national or international level by organizations from the public and private sectors for environmental and social improvements. They emerged as a new instrument of transnational trade governance [19]. Multiple VSS certifications exist in coffee productions such as Bird-Friendly, Rainforest Alliance, UTZ, Organic, Fair Trade, Coffee Practices, Appellation of Origin, and more. These certifications search to improve the economic sustainability of smallholder farmers in coffee-producing countries.

## III. BLOCKCHAIN-BASED TRACEABILITY IN AGRIFOOD PROCESSES

### A. BLOCKCHAIN BACKGROUND

According to [20], Blockchain-based traceability has a significant impact on agri-food traceability; since the information stored in the Blockchain network cannot be modified, the traceability processes will be more secure and reliable, having an impact mainly on transparency, accountability, fraud prevention, and protection among others. This makes Blockchain traceability more efficient than normal traceability processes, making it more attractive to trace product information from its origin to the final consumer. Figure 1 shows the difference between the traditional traceability and the Blockchain-based model on an agri-food supply chain.

As shown in figure 1, the agri-food supply chain layer shows the traceability stages. These partners produce and demand information related to one or multiple supply chain stages. This data is stored in a traditional centralized traceability scheme in private servers. Partners have delayed access to data of other traceable partners, so real-time data

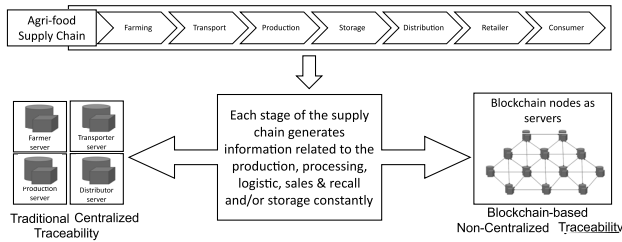


FIGURE 1. Traditional and blockchain-based traceability.

sharing has several challenges due to the different data request processes they must perform for each partner. This model influences information security and quality control. No one outside the organization can audit data, so if someone within the organization changes the data, there is a possibility that the alteration will not be detected in the following stages of the supply chain.

In Colombia, the National Institute for Food and Drug Surveillance (INVIMA) [21] has made efforts to improve the traceability of medicines and food, with the development of the *Traza* tool, a health surveillance model aimed at monitoring the conditions of establishments, products or services that are the competence of the institute throughout its entire production chain or life cycle, guaranteeing their identification and traceability. However, the tool can only be used for medicines and medical equipment and the information is centralized on the INVIMA servers. Users must access the tool through the INVIMA web portal.

This scheme has flaws because it is centralized, and small producers may not have enough technology and accessories to use this type of platform and the information it provides. This would be reflected in the level of traceability that can be applied to the products monitored by the institute. On the other hand, there are restrictions on the information that a small producer can collect based on the monetary capital, which could cause its adaptation to recent technology that the *Traza* tool may request from them, once it is available for food products. So, they might see their production costs increased just to meet the tool’s requirements.

On a Blockchain-based non-centralized traceability scheme, the data is sent to all nodes on the network. It contributes to data sharing, facilitating real-time data access and continuous process improvement on the supply chain. The transactions and all the information about the operations are open and transparent. Any traceability partner with access to the Blockchain network can see and audit all the data that has been stored. It improves data availability by not depending on a single server, increasing the security and reliability of the data stored on the different processes associated with the supply chain.

Figure 2 shows an adaptation of the architecture proposed by [20] for a Blockchain-based food traceability system. The architecture has four layers; the business layer covers all the activities related to each traceability partner on any stage of the supply chain. Any actor can consult, control and

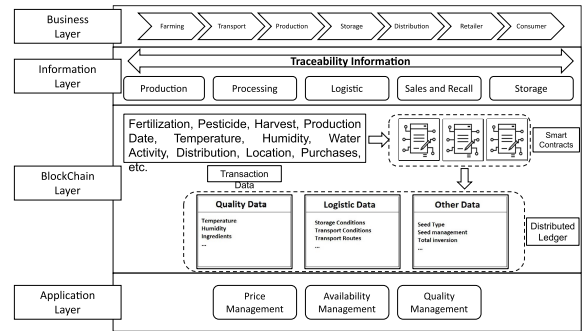


FIGURE 2. The architecture of blockchain-based food traceability system.

manage the traceable information. Information layer process all data produced by the traceability partners. It includes quality information, processing data, logistic information, and other data types. This data can be recorded using IoT sensors, non-IoT sensors, or manually.

The Blockchain layer facilitates transparency and data security. The smart contracts monitor the quality of the product and planning. It provides retailers and consumers a record of how the product has moved through the supply chain, all this in real-time using the information appended on each transaction and the information already stored in the Blockchain network. Finally, the application layer will see the interaction between traceability partners. They would be able to see all the activity record on the full process.

**B. CURRENT KNOWLEDGE STATUS**

Considering the investigation aim and the previous concepts, we conducted a literature review to improve the understanding of the subject and see the current state of the application of Blockchain technology to increase data reliability or perform data validation in traceability environment and its use during coffee transport or storage.

The Authors in [22] presents a system for the preservation of evidence based on Blockchain technology to avoid problems caused by third-party evidence preservation services. Still, it is not covered from the point of view of the reliability of the evidence before it is stored on the network or the reliability of who holds the evidence. In [23], the authors present an approach using Blockchain technology in the food supply chain, whose goal is to increase standardization in the data format, improve the lack of regulations, etc. Nevertheless, they do not consider the reliability of the data to be appended to the network, leaving the entire validation process to an actor of the Blockchain network.

The authors in [24] present a reliability analysis for Blockchain oracles using Fault Tree Analysis based on the oracle architecture. Although the authors evaluate the reliability of the Blockchain oracle, oracles introduce a “trusted” third party into the Blockchain system. But the external state used for oracle data can be altered.

In [25], the author proposed Blockchain to increase transparency, sustainability, scalability, safety, security, and

accuracy on a real-time, cost-effective coffee supply chain. However, the authors only propose using the technology but do not perform tests or simulations in the traceability of the coffee supply chain. In [26], the authors show the typically established smart contracts between parties in an intelligent farming environment. All the contracts aim to improve traceability transparency, but none consider data monitoring or validation.

Authors in [27] monitored green coffee beans during transport from Brazil to Italy to measure the effect on green coffee beans by changes in temperature, relative humidity, and bean moisture to determine the best disposition for the transport of coffee in containers and generate recommendations in this regard. However, the suggestions made by the authors can be applied on the coffee bags before starting the transport process or at the end of it, leaving aside adjustments that could be made during the transport process. In [28], the authors propose a temperature and humidity control for the storage of Arabica coffee by controlling the activation and deactivation of a fan located in the storage area to maintain control over the mentioned variables. Still, the temperature and humidity control are performed over the entire storage area, leaving aside the conditions of coffee beans, which could affect the beans in good conditions.

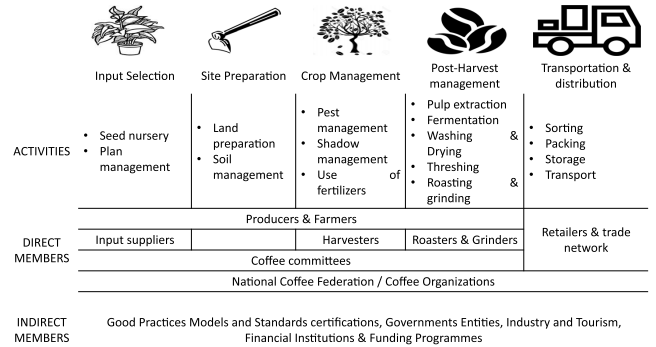
In [29], the authors proposed an IoT-enabled smart agriculture system using Blockchain technology for the pre-harvest and post-harvest stages. Nevertheless, the proposed system does not consider that the data could be damaged or tampered before the transaction over the network is done, although they consider it a challenge.

**C. POSITIONING AND CONTRIBUTION**

Based on the previous sections, we can see the following facts.

1. The localized investigations related to Blockchain technology have focused on using the advantages offered by the technology, especially its immutability, to improve the transparency of traceability processes. Still, no efforts have been located in the use of this to validate the information before being attached to the network Blockchain. This could allow the entry of altered data to the network that cannot be modified.
2. It was possible to locate research efforts focused on monitoring the most relevant variables for the conservation of the quality characteristics of the coffee, the control of these variables, and the effects that they have on the production processes.
3. Even though some authors mention the use of traceability schemes supported in Blockchain, to improve the transparency of the information. They do not mention the use of smart contracts to validate the data generated in each stage of the supply chain, or the generation of automatic recommendations based on data that support early decision-making

To monitor variables related to the quality of the coffee, in this paper, we propose a smart contract for transporting and



**FIGURE 3. General coffee supply chain based on [30].**

storing coffee bags. Using the monitored data, it is possible to verify the handling given to each bag and made a recommendation about the purchase price. The proposed smart contract validates the data sent to the Blockchain network in each transaction. These can be used in the future to measure reliability in the information generator, giving new validation rules and building even more confidence in traceability data.

**IV. SMART COFFEE CHAIN**

This section presents the coffee supply chain and the smart contract to coffee storing and transportation.

**A. COFFEE SUPPLY CHAIN**

The coffee supply chain has five main stages, as shown in Figure 3. Each step involves a set of specific activities [30]. Direct and indirect actors of each stage conduct these activities. The size of the chain depends on the size of the farm, the processes that occur in the farm, and the customer preferences, among other considerations.

Figure 3 shows the general coffee supply chain with its actors and activities. The first stage is the input selection; at this stage, the farmer selects the coffee seed to use based on the recommendations made by the coffee committees, national coffee committees, or coffee organizations. Then the input suppliers provide the seeds to the farmer and the handling required for the seed. In the second stage, the farmer is responsible for preparing the soil and managing it. He will use various chemical products according to the earth’s needs and the seed. Again, as direct actors, we also have committees and coffee organizations or federations.

In the third stage, the culture management process is conducted; in this case, diseases associated with coffee, shadow management should be controlled, if required, the use of fertilizers to maintain the plant healthy and prune weeds or other types of unwanted plants. Here, the harvesters, responsible for collecting the fruit, appears, a new direct actor. Stage 4 consists of the handling of the fruit; this includes the process of pulping, fermentation, washing, drying, toasting, etc. In this stage, a new actor appears; the roasters and mills, which have specific tasks within the scene, although farmers can possess the means to conduct these activities.

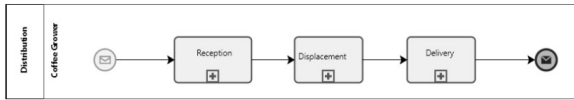


FIGURE 4. Distribution processes on the coffee supply chain.

Post-harvest processes, like storage and transportation, are affected by environmental conditions; humidity affects product quality in these processes.

Finally, the last stage is the transportation and distribution of the coffee; two of the activities associated with this stage may be present in the previous stages of the chain. The storage and transport can occur internally on a farm or externally to transfer the coffee to a collection center with greater storage capacity.

As can be seen in figure 4, the Transportation starts with the coffee bag reception. Then the bags are taken to the destination point, where they are delivered to the corresponding stage of the chain. In this way, our proposed smart contract will focus on the displacement stage. At this stage, the coffee must be monitored to ensure that it is in a good state upon delivery.

The storage process can be seen in figure 5. The coffee bags are organized by lots and stored until enough bags are collected. Then the coffee is classified, cleaned, and milled to be finally classified by size, weight, density, and color. Finally, it is packed and stored to be sold.

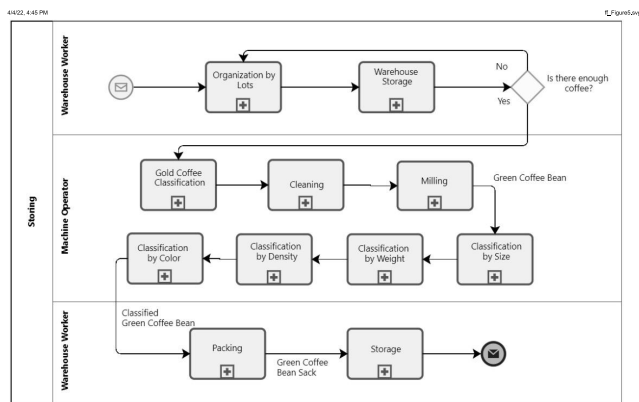


FIGURE 5. Storage processes on the coffee supply chain.

In this paper, the Storing will be focused on the warehouse storage of green coffee bags. This stage is essential because the quality coffee beans are susceptible to variables like bean moisture, temperature, and water activity. It is valid to mention that the loss of characteristics and quality in coffee is a big problem; since 100 Kilograms of cherry coffee are needed to obtain 20 Kgs of dry parchment coffee. From this, 18 Kilograms are received in the best of cases of green coffee. After the roasting process, 13 Kilograms of roasted coffee are obtained. For every 100 Kilograms of coffee harvested, we will get 13 Kilograms of coffee finally commercialized.

Herein lies the importance of maintaining optimal coffee conditions during Transportation and Storing in warehouses.

**B. SMART CONTRACT TO COFFEE STORING AND TRANSPORTATION**

Smart contracts record all information generated by the actors on the five main processes from input selection to transportation and distribution. All actors would be able to retrieve all the information related to a specific activity at any time during the product flow on the supply chain. Once the information is uploaded into the ledger on the Blockchain network, no one will be able to manipulate it in any way. Next, we show the use of the smart contract on Storing and Distribution activities in Transportation & distribution processes.

Table 1 shows the data available for the smart contract to Storing and Transportation activities in Transportation & distribution processes. At each stage, a predefined smart contract between the involved actors will manage the data and will be able to propose decisions to the actors based on the transaction’s information. The predefined smart contract will be developed using the initial arrangement made by the network actors.

TABLE 1. Data available for the smart contract by activity of interest.

Activity (based on Fig. 3)	Data Traced	Members
Storage	Packaging, Temperature, Humidity, Gran moisture, Number of bags, Storing time, Origin.	Producers & Farmers – Retailers & Trade network
Transport	Packaging, Temperature, Humidity, Gran moisture, Number of bags, Origin, Destination, Transport type, Transport time.	

A network actor could access multiple smart contracts depending on business relationships or inquire about the coffee’s status at each stage. We decided to include Producers & Farmers because transport and storage can also occur at the post-harvest management stage.

Our goal is that the smart contract can check every transaction made into the system and perform recommendations about the product management and penalties based on the predefined terms agreed by the chain actors; following this, Figure 6 shows the minimal required stages to build and deploy a smart contract.

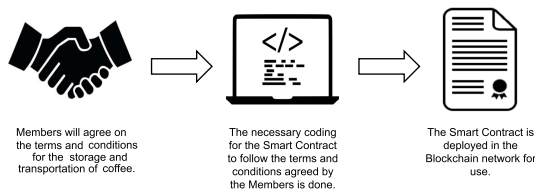


FIGURE 6. Smart contract development stages.

In figure 6, the coffee supply chain actors of the storage and transport activities will be the contract owners; they will agree to the terms and conditions for the initial development of the smart contract. Once the negotiation is done, the smart contract programmer will develop the contract considering all the agreed points like prices per bean quality and classification. Then the smart contract programmer will deploy the contract on the Blockchain network, and actors connected to the network, with the required permissions, will conduct transactions using the deployed smart contract.

Additionally, the developed smart contract will include conditionals that will help to determine when incorrect data has been received in a transaction. Validations initially will look for syntax errors, but we expect that future interactions will detect semantic errors in the transactional data.

The codification of a smart contract requires a set of attributes, functions, and events that triggers the transaction log for a new block on the Blockchain network. The attributes are the variables that will hold the transactional data and any data generated during the contract execution. The functions represent the smart contract’s task once a network transaction calls it. Finally, the events occur once the smart contract has finished and require updating an asset on the Blockchain network.

The smart contract will bridge the real world and the Blockchain network, collecting data and updating the traceable variables logs. We deploy a test Blockchain network using the Hyperledger Fabric to evaluate our proposed smart contract. The test network consists of two organizations with two and three transactional endpoints known as Peers; the peers will be in charge of sending the transaction to the network on the dedicated channel; an organization can be a multinational corporation, a national company or a single individual. In Hyperledger Fabric the channels are the way of communication for peers; a peer can join multiple channels at the same time, each transaction made by a peer must have a target channel and smart contract.

Each peer has a unique identity that helps them verify their signatures over any transaction and the authority or access level on each channel. Peers can do queries over the store data but cannot create or update the status of a variable independently. To update a variable status, all the peers on the network must reach a consensus; the test network uses RATF [31] as consensus protocol; once the consensus is reached, the ledger is updated with the new status.

Figure 7 shows the general system for deploying and using the smart contract to coffee storing and transportation. At the top, we can see the smart contract programmers in charge of the contract deployment over the Blockchain network; we will be using the Hyperledger Fabric Blockchain solution. Also, we can see that the actors involved with the contract will see the transaction information as soon as it is available on the Blockchain network.

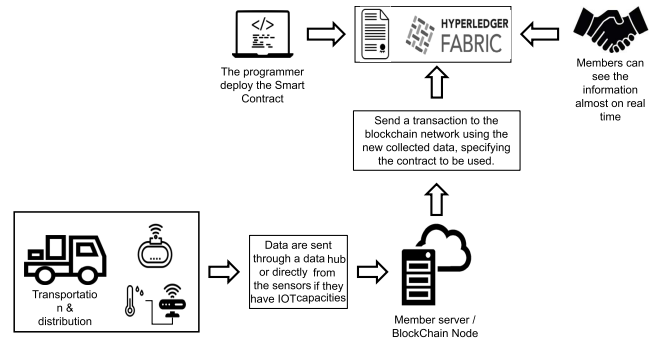


FIGURE 7. General system overview.

Also, in Figure 7 we can see the stage of the coffee supply chain (figure 3) in which we will have our research focus, Transportation & Distribution. The activities and information available can be seen in table 3; the data in most cases will be collected using sensors connected to a data hub or with IoT capabilities. The data cannot be collected automatically by sensors. It will be done manually and stored in digital media to be sent to the respective actor server, which will also work as a node of the Blockchain network. Finally, the server will send a transaction into the individual channel using the latest data and the smart target contract using one of its peers.

Figure 8 shows the interaction with the Blockchain network; on the left side, we can see that the interactions start with the traceable data generated by coffee transport and storage; this information could be recollected using sensors. Later, this traceable data is sent to the actor’s server. Using the Hyperledger Fabric, the deployed smart contract updates the states of the information log on the Blockchain network. On the other hand, the supply chain actors can consult the

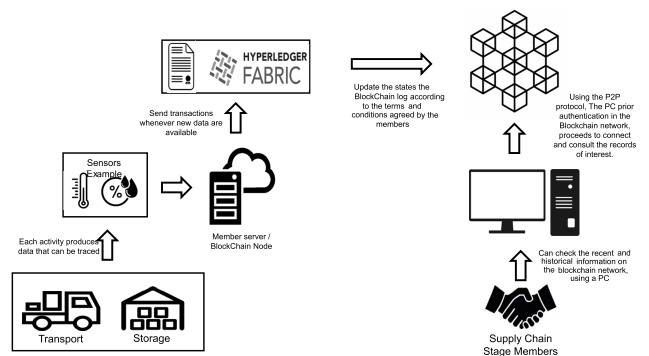


FIGURE 8. Interaction with the blockchain network.

Blockchain network's information log using a PC that has previously acquired the necessary credentials; to connect to the Blockchain channel, and to be able to query the information log. This process, if necessary, could also be extended to actors who are not part of these activities.

Our proposal implements a smart contract written in JavaScript, one of the multiple programming languages supported by the Hyperledger Fabric. The contract can be used for storage and transport. The smart contract will know what activity will be traced at the initial deployment. The proposed smart contract can self-check the transactional data looking for syntax errors in all cases. Also, it will compare the newly entered data with the optimal values agreed by actors to check if any penalty or recommendation should be applied. All these penalties and recommendations will be stored in the Blockchain to be verifiable by current and future users or used as quality measurements.

## V. IMPLEMENTATION AND PERFORMANCE EVALUATION

As mentioned, we selected JavaScript to write our proposed smart contract for experimental and evaluation purposes. To evaluate the smart Contract, we use Hyperledger Caliper. This tool lets us see metrics such as throughput, latency, successful and failed transactions on the deployed Blockchain network. Based on [32], we will focus only on these Blockchain challenges to ensure that the proposed smart contract will not threaten the network.

Caliper throughput metric is the rate at which the Blockchain commits valid transactions in the defined time window. Latency metric is defined as a network-wide view of the amount of time taken for a transaction's effect to be usable across the network and the amount of successful and failed transactions.

We use a remote access virtual machine running 64 bits ubuntu 16.04 LTS with four cores, 8 GB of RAM, and 200 GB of disk space to install all the necessary tools to deploy the Hyperledger Fabric network and Hyperledger Caliper.

Because coffee bags area is evaluated randomly a couple of times every week for bean moisture, and for ambient temperature and humidity every month, we decided to run a stress test using Caliper with 30 to 600 seconds duration to change the transaction load. This behavior will not replicate the expected natural behavior but will let us see if the deployed Contract will endanger the stability of the network.

Table 2 shows the most critical parameters to configure our experimental setup; other parameters depend on the configured Blockchain network, such as the number of peers, orderers, channels, and organizations. The test duration was selected randomly, starting with 30 seconds and then modifying the configuration file to reach 600 seconds using 30-second increments during intervals. The number of Workers and the maximum transactional load was selected after performing multiple tests. Using the results, we determine that with the computational capacity of the virtual machine, exceeding the value of 6 workers and 10 as the transaction load supposed a load that exceeded the virtual machine

TABLE 2. Hyperledger caliper configuration.

Variable	Description	Value
TxDuration	Duration of the test in seconds.	30 – 600 (Increased using steps of 30 seconds)
Workers	Number of clients sending transactions on the network	6
TransactionLoad	The number of transactions being processed by the network that is to be maintained	2 or 10

capacity, decreasing the number of successful transactions and increasing the number of failed transactions as the test extended over time, using the Fixed Load Rate Controller on the Caliper Configuration. This controller will maintain a defined backlog of transactions within the system by modifying the driven TPS. The result is the maximum possible TPS for the system while retaining the pending transaction load.

Next, we present pseudocode and a Finite State Machine (FSM) of the essential algorithms in the smart contract. Algorithm 1 shows the smart contract initialization. This algorithm creates the *Bag0* asset and stores it on the ledger. This asset saves all the optimal parameters agreed by the actors. Figure 9 shows the FSM where ScI is the smart contract initialization, PS is the status update on the ledger, GS is the status check on the ledger and AU is the asset update function.

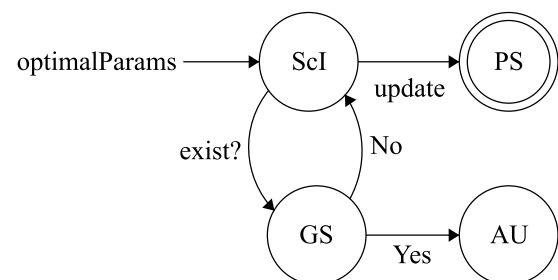


FIGURE 9. Algorithm 1 FSM.

Algorithm 2 shows the asset creation process. This process is like the smart contract initialization but in this case the algorithm uses the transactional data related to an actual Bag, utilizing an asset ID that is a bag id identifier like Bag1 or Bag27. Each time a coffee bag will be added to the traceability Blockchain network, this algorithm is called. Figure 10 shows the FSM where AC is the asset creation function, errors is the syntax error detection function, pen is the penalties generation functions and reco is the

**Algorithm 1** Smart Contract Initialization

**Inputs:**  
 ID: key value of a product  
 POI: Product optimal information

**Outputs:**  
 product = GetState(ID);  
**if** product == null **then**  
     PutState(key, POI, “none”, “none”, “none”, 1.0, 1.0);  
**else**  
     assetUpdate();  
**end if**  
 Initial asset created with agreed optimal parameters

**Algorithm 2** Smart Contract Asset Creation

**Inputs:**  
 ID: key value of a product  
 PSI: Product sensed information

**Outputs:**  
 product = GetState(ID);  
**if** product == null **then**  
     errors = syntaxVal(PSI);  
     reco = recoGen(PSI);  
     penalty = penaltyGen(PSI);  
     PutState(key, PSI, errors, reco, penalty);  
**else**  
     assetUpdate();  
**end if**

Asset created with the traced data or updated with the traced data, syntax errors located, recommendations, and price penalties.

**Algorithm 3** Smart Contract Asset Update

**Inputs:**  
 ID: key value of a product  
 PSI: Product sensed information

**Outputs:**  
 product = GetState(ID);  
**if** product != null **then**  
     errors = syntaxVal(PSI);  
     reco = recoGen(PSI);  
     penalty = penaltyGen(PSI);  
     PutState(key, PSI, errors, reco, penalty);  
**else**  
     assetCreate();  
**end if**

Asset updated or created with the traced data, syntax errors, recommendations, and price penalties.

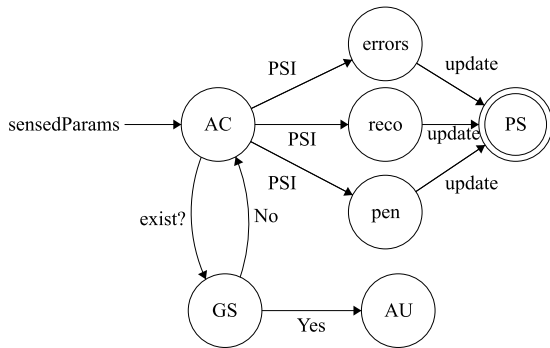


FIGURE 10. Algorithm 2 FSM.

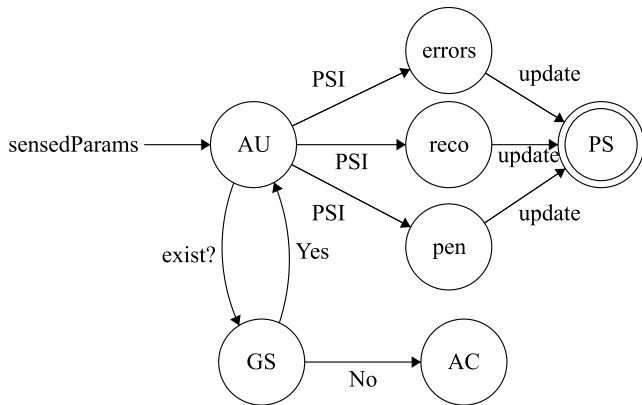


FIGURE 11. Algorithm 3 FSM.

recommendation generation function. If the asset already exists, the machine state transitions to the update state.

Algorithm 3 shows the asset update process. This algorithm updates asset states on the ledger; as previously mentioned, each coffee bag has an ID stored as an asset.ID. Using this ID, the smart contract updates the state of all the bag data on the ledger. This algorithm calls algorithms 4 – 6 to check for syntax errors on the transactional data, set penalties, and management recommendations. Figure 11 shows the FSM similar to figure 10; the update function states use the previously create asset to generate penalties and recommendations, if the asset is not created, the machine state goes to create asset.

Algorithm 4 shows the syntax error check; in this case, the smart contract checks if the sensed data can be converted from string to numeric data. If the data cannot be, the algorithm returns the data value stored on the ledger as corrupted data to be analyzed later by a warehouse operator or the transport agency. If the data can be converted, the algorithm returns the string *none*. Figure 12 shows the FSM where SynChk is the syntax error check function that returns a string containing all the errors found.

Algorithm 5 shows the penalty check process. In this case, the smart contract compares the newest coffee bag data with the *Bag0* data; as we mentioned before the asset identified as *Bag0* stores the optimal parameters agreed by the actors. If the coffee bag parameters do not correspond with the optimal values a penalty should be applied according to the agreed terms. Figure 13 shows the FSM where GSOp is the optimal agreed parameter asset locator, DC is the data comparison function, and penal is the penalty generation function.



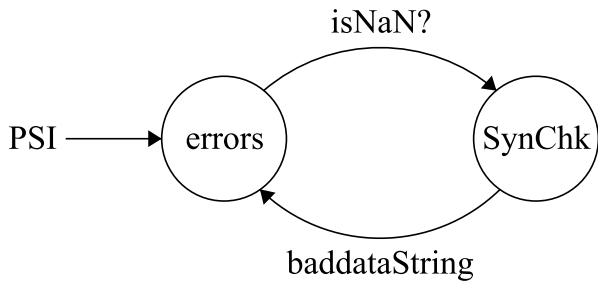


FIGURE 12. Algorithm 4 FSM.

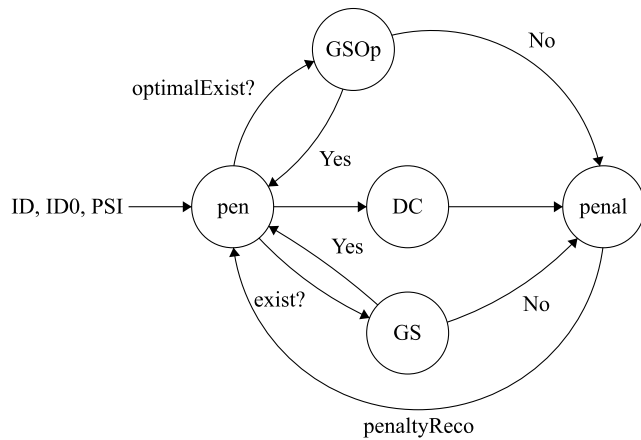


FIGURE 13. Algorithm 5 FSM.

Finally, algorithm 6 shows the smart contract recommendation function. The process is similar to the penalty algorithm. But in this case, the smart contract will generate a management recommendation to avoid the penalty and ensure the quality of the coffee bag. Figure 14 shows the FSM where GHS is the historical status check function, retrieves all the relevant information about the asset from the ledger.

**Algorithm 4** Smart contract Syntax Error Check

```

Input:
PSI: Product sensed information
Outputs:
baddataString = "";
for data in PSI
  if (!isNaN(data)) then
    baddataString = baddataString + "none";
  else
    baddataString = baddataString + string(data);
  end if
end for
return baddataString;
Syntax error detected on the product sensed information
    
```

After developing the smart contract and using table 2 parameters, we perform the stress test over the network in two stages. In the first stage the Caliper management will send the workers to create bags on the Blockchain network

**Algorithm 5** Smart Contract Penalty Check

```

Input:
ID: key value of a product
ID0: key value of the optimal product condition
PSI: Product sensed information
Outputs:
Penal = "";
product = GetState(ID);
productOp = getstate(ID0);
if product != null & productOp != null then
  dataOp = productOp.PSI;
  data = product.PSI
  if data != dataOp then
    penal = penal + "Penalty recommendation based on
    nonoptimal parameter";
  end if
end if
return penal;
Penalty or penalties to be applied
    
```

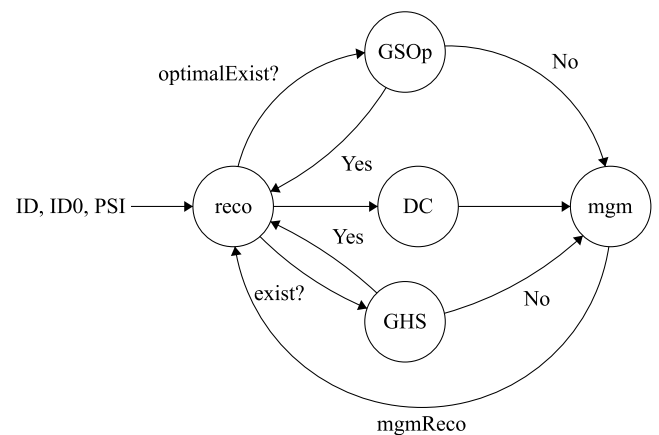


FIGURE 14. Algorithm 6 FSM.

(2 or 10 for workers depending on the transaction load). Later, the workers will randomly send read transactions for each created bag to ensure that all bags get a read transaction from a different worker. The second stage will evaluate the update function that will run algorithms 1-4 to assess the bag condition; generate recommendations, and penalties. In this stage, the manager will also send the workers to create bags on the Blockchain network, and the worker will send update transactions randomly for every bag created. The results of these processes can be seen in Figures 15-22.

Figure 15 showed the number of successful transactions when the task was just to read the information of a coffee bag. In this case, there was no failed transaction on the test. However, in Figure 16, we can see that the number of successful transactions decreased drastically on each run and that the number of failed transactions increased over time. To address this, we debugged the algorithms 3-6; after that, we confirmed that the decreased successful transactions and

**Algorithm 6** Smart Contract Recommendation

**Input:**

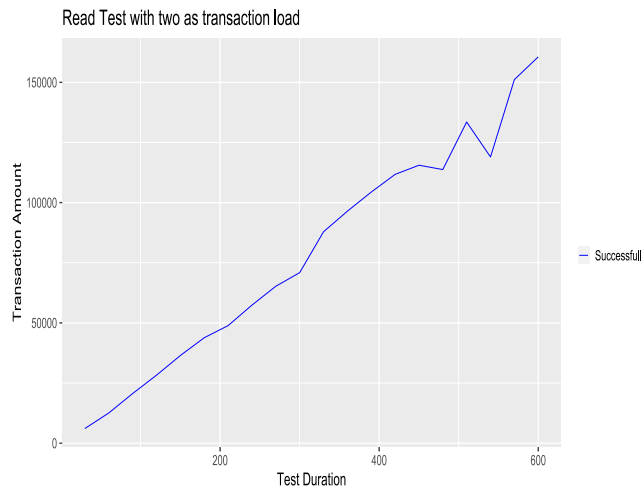
- ID: key value of a product
- IDO: key value of the optimal product condition
- PSI: Product sensed information

**Outputs:**

```

reco = ""
productOp = getState(IDO);
productHistory = GetHistoricalState(ID);
if productHistory!= null & productOp != null then
  for product in productHistory
    dataOp = productOp.PSI;
    data = product.PSI
    if data != dataOp then
      reco = reco + "expert management
      recommendation based on nonoptimal parameter";
    end if
  end for
end if
return reco;
    
```

Management recommendation



**FIGURE 15.** Hyperledger caliper read results using two as transaction load.

the increased failed transactions were not due to a bug or problem in the smart contract developed.

However, after reviewing the Log of the Caliper tool, we realized that the problem is presented because two or more workers try to update the same bag at the same time or when the transaction order isn't correct according to [33], Hyperledger Fabric uses a single trustworthy ordering service in charge of the transaction order. So, if a transaction order sets an update using a status previously updated but not reflected on all the nodes the transaction will be unsuccessful. Therefore, only the first transaction that arrived at the network is processed, and the other ones became unsuccessful transactions. After reviewing the viable solutions to this problem, the most viable for this case suggested the modification of the smart contract to support a list of transactions. However,



**FIGURE 16.** Hyperledger caliper update results using two as transaction load.

this solution would be to the detriment of the number of transactions performed per second on the network.

Bearing this in mind and considering that in a real environment, the chances of receiving at the same time multiple transactions related to the same bag are meager, it was determined not to modify the previously developed smart contract.

Figure 16 and figure 17 show the Throughput and Average latency differences while sending the read and update transaction to the Blockchain network. While sending read transactions, the throughput is high, an average of 246 transactions per second. It is sufficiently fast enough to respond to the environment where the smart contract would be running. These values allow us to conclude that Multiple Warehouse Operators could make inquiries about the network about the state of multiple coffee bags in parallel without causing a significant burden on it.

However, to update transactions while executing the smart contract, the throughput falls to an average of approximately ten transactions per second. In this case, considering the update window used in the consulted storage sites, the transaction rate per second remains sufficient to carry a control of the stored or transported coffee bags.

In the case of average latency, we can see in Figure 18 that the latency is constant (0.02 seconds) for the reading stage. At the same time, for the case of a update, this value goes up and remains at an average of 0.5 seconds. However, a considerable increase in latency remains manageable within the field of application, considering the number of successful transactions made in each cycle (Figure 16) of updates.

As shown in Figure 19, the behavior previously evidenced is maintained with a smaller load of transactions, achieving a more considerable number of successful transactions without any errors, confirming the previous findings. In the case of update transactions (Figure 20), we can see a drop in the number of transactions conducted successfully between the cycles of 300 and 390 seconds. When reviewing the Log information of the Caliper tool, we could not find an explanation to this anomalous behavior, so we attributed its cause to a fall in the

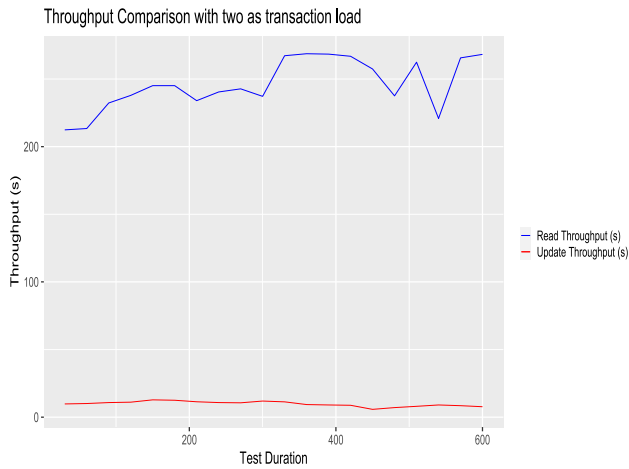


FIGURE 17. Throughput comparison between the read and the update operations.

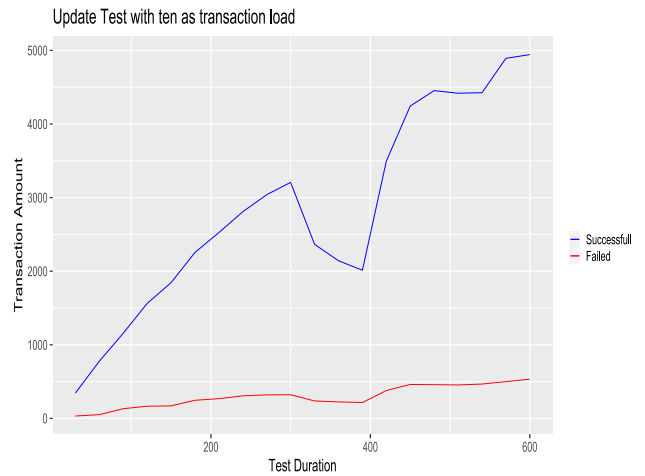


FIGURE 20. Hyperledger caliper update results using ten as transaction load.

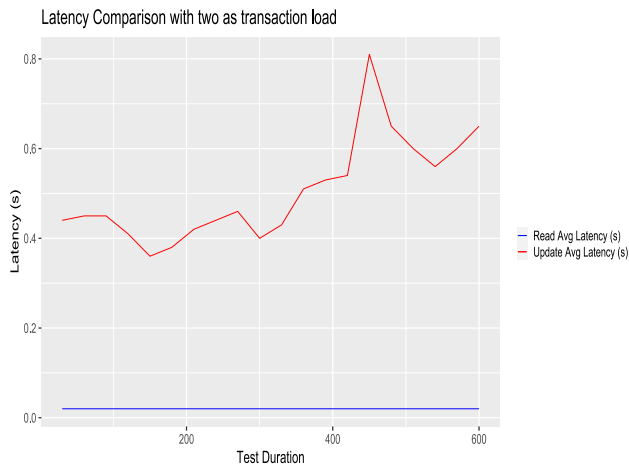


FIGURE 18. Average latency comparison between the read and update operations.

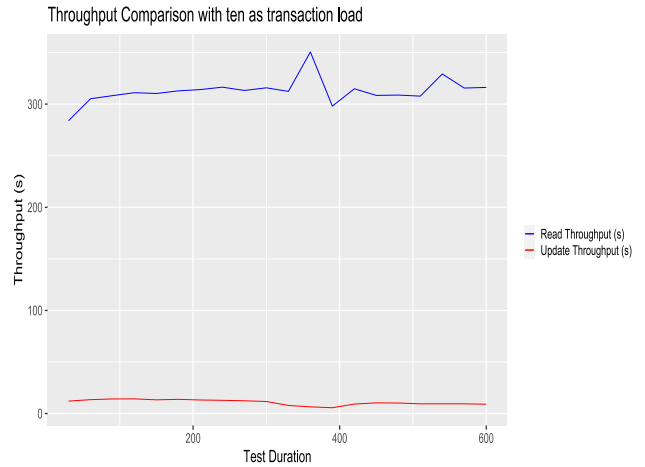


FIGURE 21. Throughput comparison between the read and the update operations.

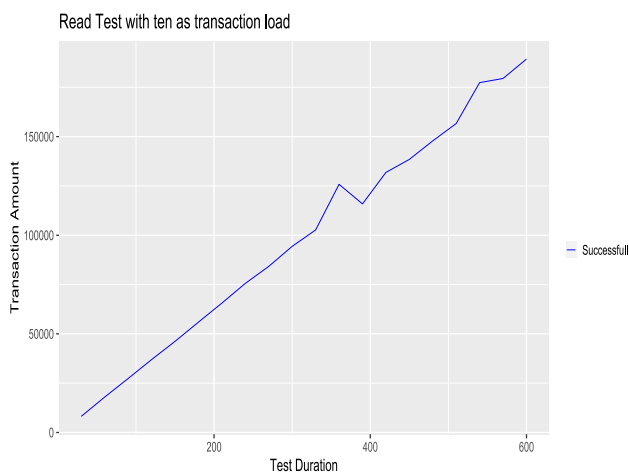


FIGURE 19. Hyperledger caliper read results using ten as transaction load.

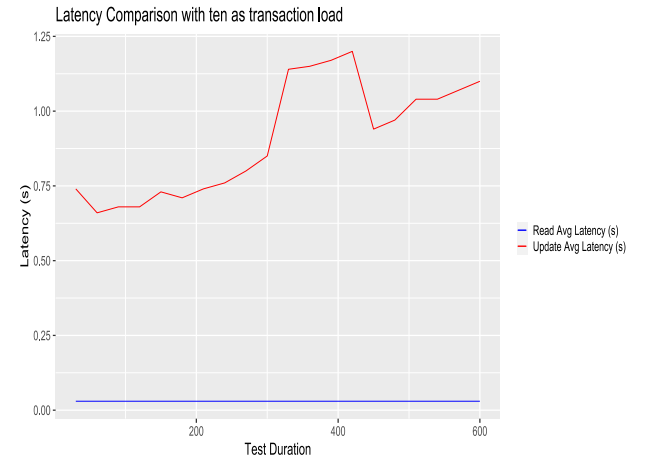


FIGURE 22. Average latency comparison between the read and update operations.

performance of the virtual machine in which the Blockchain network deployment was made.

However, observing the values achieved, we can consider that in an actual application of our smart contract, the number

of successful transactions is sufficiently high to meet the conditions of current measurements in coffee bags, based on the consulted local sources, according to the warehouse operator, they check the coffee conditions a couple of times

every week. Again, it is valid to mention that failed transactions are because it was attempted to perform more than one transaction related to the same coffee bag in parallel.

It can be seen in Figure 21 and Figure 22 that the behavior for the reading and update cycle is similar to the previous case evaluated. However, due to the update stage between the 300- and 420-Seconds processes, latency increases; it goes hand in hand with the fall in successful transactions. We can confirm that this drop in transactions is due to an event related to the system in which the test network was run and not with the smart contract.

## VI. CONCLUSION

Technologies like Blockchain, IoT, and others are improving how some tasks related to supply chains are developed. Tasks associated with the traceability processes are conducted quickly and automatically or semiautomatically. Information is readily available to everyone involved in the supply chain. All information is kept secure and immutable in the chains that use Blockchain technology, having the possibility of quickly consulting the status of a product in different links in the chain at any time, without consulting a third.

The proposed smart Contract and the traceability scheme with Blockchain technology, using expert knowledge, will perform timely action recommendations to reduce the loss of coffee sacks due to improper handling during transport or storage in warehouses. All these recommendations will be made automatically based on the collected data and notified every user involved. In this way, we would have more bags complying with the parameters stipulated by the FNC, receiving a better price for the product, increasing the coffee grower's profit. All this without interfering with the current prices based on the different certifications available.

Also, data validation will improve the reliability in this type of technology applied to supply chains of precious commodities like coffee reducing measurement error. Additionally, the recommendations made can be consulted at any moment in the Blockchain network. These will allow those involved in the supply chain to make quick decisions about coffee management, which is about to arrive or the coffee in storage when required.

The use of this scheme will facilitate data access. Due to the Blockchain network decentralized feature, all the historical and recent data will be available to all the users with access to the Blockchain network at any moment and in real-time. The integrity of the data will be guaranteed by the intrinsic characteristics of the Blockchain network and enhanced by the data evaluation stage conducted by the smart contract. The data transparency will also be increased by being available throughout the flow process at the supply chain stages.

So, more transparency, security, and reliability on the data will be acquired. Also, the efficiency will be improved with the decision support and the profit increase. Finally, this smart contract and the Blockchain-cased traceability scheme will not invalidate any Coffee Certifications or VSS used by the coffee grower, the coffee exporter, or roaster.

Although the initial settings cost of the sensors, the Blockchain network, the development and deployment of the smart contract could be high, the benefits of this technology in traceability environments offer automated and robust processes that will lead to better, more reliable traceability system.

However, the proposed smart Contract is still in an early development stage. More variables and knowledge will be needed to make it more robust and practical to use on another stage of the coffee supply chain. More work is required to make sure it is feasible and has practical significance, and future iterations will include validations rules not considered at this point. In-situ validations are needed to find problems not presented in controlled environments.

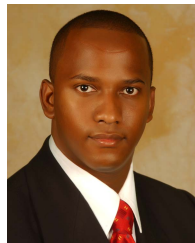
## ACKNOWLEDGMENT

The authors are grateful to the Telematics Engineering Group (GIT) and the Research Group for Rural Development (Tull) of the University of Cauca, and the Ministry of Science Technology and Innovation of Colombia (Minciencias) for the Ph.D. support granted to Cristian Valencia-Payan.

## REFERENCES

- [1] N. Bhumiratana, K. Adhikari, and E. Chambers, "Evolution of sensory aroma attributes from coffee beans to brewed coffee," *LWT Food Sci. Technol.*, vol. 44, no. 10, pp. 2185–2192, Dec. 2011, doi: [10.1016/j.lwt.2011.07.001](https://doi.org/10.1016/j.lwt.2011.07.001).
- [2] H. T. T. Truong, "Community initiatives and local networks among K'ho cil smallholder coffee farmers in the central highlands of Vietnam: A case study," *J. Asian Afr. Stud.*, vol. 55, no. 6, pp. 880–895, Sep. 2020, doi: [10.1177/0021909620935427](https://doi.org/10.1177/0021909620935427).
- [3] T. Dietz, A. Estrella Chong, J. Grabs, and B. Kilian, "How effective is multiple certification in improving the economic conditions of smallholder farmers? Evidence from an impact evaluation in Colombia's coffee belt," *J. Develop. Stud.*, vol. 56, no. 6, pp. 1141–1160, Jun. 2020, doi: [10.1080/00220388.2019.1632433](https://doi.org/10.1080/00220388.2019.1632433).
- [4] J. G. Bray and J. Neilson, "Reviewing the impacts of coffee certification programmes on smallholder livelihoods," *Int. J. Biodiversity Sci., Ecosyst. Services Manage.*, vol. 13, no. 1, pp. 216–232, Jan. 2017, doi: [10.1080/21513732.2017.1316520](https://doi.org/10.1080/21513732.2017.1316520).
- [5] E. Blackmore, J. Keeley, R. Pyburn, E. Mangus, L. Chen, and Q. Yuhui, "Assessing the benefits of sustainability certification for small-scale farmers in Asia," in *Natural Resources Issues*, no. 25. London, U.K.: IIED.
- [6] F. F. N. de Cafeteros. *Federación Nacional de Cafeteros*. Accessed: Jan. 8, 2021. [Online]. Available: <https://federaciondefeteros.org/wp/>
- [7] F. F. N. de Cafeteros. (2016). *Resolución 02 de 2016*. Accessed: Jan. 8, 2021. [Online]. Available: <https://federaciondefeteros.org/app/uploads/2019/11/6.-Normas-de-calidad-para-caf?-verde-en-almendra-para-exportaci?n-Resoluci?n-2-2016.pdf>
- [8] F. F. N. de Cafeteros. (2021). *Aprenda a Vender su café*. Accessed: Jan. 8, 2021. [Online]. Available: <https://federaciondefeteros.org/wp/servicios-al-caficultor/aprenda-a-vender-su-cafe/>
- [9] J. Lernoud, J. Potts, G. Sampson, V. Voora, H. Willer, and J. Wozniak, "The state of sustainable markets 2017," ITC, Geneva, Switzerland, 2017.
- [10] F. Tian, "An agri-food supply chain traceability system for China based on RFID & blockchain technology," in *Proc. 13th Int. Conf. Service Syst. Service Manage. (ICSSSM)*, Jun. 2016, pp. 1–6, doi: [10.1109/ICSSSM.2016.7538424](https://doi.org/10.1109/ICSSSM.2016.7538424).
- [11] I.-H. Hong, J.-F. Dang, Y.-H. Tsai, C.-S. Liu, W.-T. Lee, M.-L. Wang, and P.-C. Chen, "An RFID application in the food supply chain: A case study of convenience stores in Taiwan," *J. Food Eng.*, vol. 106, no. 2, pp. 119–126, Sep. 2011, doi: [10.1016/j.jfoodeng.2011.04.014](https://doi.org/10.1016/j.jfoodeng.2011.04.014).
- [12] C. Sánchez, G. Schneider, and M. Leucker, "Reliable smart contracts: State-of-the-art, applications, challenges and future directions," in *Leveraging Applications of Formal Methods, Verification and Validation. Industrial Practice (Lecture Notes in Computer Science)*, vol. 11247. Cham, Switzerland: Springer, 2018, pp. 275–279, doi: [10.1007/978-3-030-03427-6\\_21](https://doi.org/10.1007/978-3-030-03427-6_21).

- [13] N. Szabo. (1996). *Smart Contracts: Building Blocks for Digital Markets*. Accessed: Jan. 23, 2020. [Online]. Available: [http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart\\_contracts\\_2.html](http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html)
- [14] *EIP-190: Ethereum Smart Contract Packaging Standard*. Accessed: Oct. 25, 2021. [Online]. Available: <https://eips.ethereum.org/EIPS/eip-190>
- [15] B. Hu, Z. Zhang, J. Liu, Y. Liu, J. Yin, R. Lu, and X. Lin, "A comprehensive survey on smart contract construction and execution: Paradigms, tools, and systems," *Patterns*, vol. 2, Feb. 2020, Art. no. 100179, doi: [10.1016/j.patter.2020.100179](https://doi.org/10.1016/j.patter.2020.100179).
- [16] *ISO9000 Sistemas de Gestión de la Calidad—Fundamentos y Vocabulario*, I. Int. Org. Standardization, Geneva, Switzerland, Feb. 2015, p. 35.
- [17] K. M. Karlsen and P. Olsen, "Problems and implementation hurdles in food traceability," in *Advances in Food Traceability Techniques and Technologies: Improving Quality Throughout the Food Chain*. Cambridge, U.K.: Woodhead, 2016, pp. 35–46.
- [18] M. Di Zio, N. Fursova, T. Gelsema, S. Gießing, U. Guarnera, J. Petruskienė, L. Q. Kalben, M. Scanu, K. O. ten Bosch, M. van der Loo, K. Walsdorfer. (Jun. 2016). *Methodology for Data Validation 1.0 Essnet Validat Foundation*. Accessed: Apr. 26, 2021.
- [19] *Voluntary Sustainability Standards, Trade and Sustainable Development—3rd Flagship Report of the United Nations Forum of Sustainability Standards (UNFSS)*, UNCTAD, Geneva, Switzerland, 2018.
- [20] H. Feng, X. Wang, Y. Duan, J. Zhang, and X. Zhang, "Applying blockchain technology to improve agri-food traceability: A review of development methods, benefits and challenges," *J. Cleaner Prod.*, vol. 260, Jul. 2020, Art. no. 121031, doi: [10.1016/j.jclepro.2020.121031](https://doi.org/10.1016/j.jclepro.2020.121031).
- [21] (2014). INVIMA. *Instituto Nacional de Vigilancia de Medicamentos y Alimentos*. Accessed: Dec. 15, 2021. [Online]. Available: <https://www.invima.gov.co/>
- [22] M. Wang, Q. Wu, B. Qin, Q. Wang, J. Liu, and Z. Guan, "Lightweight and manageable digital evidence preservation system on bitcoin," *J. Comput. Sci. Technol.*, vol. 33, no. 3, pp. 568–586, May 2018, doi: [10.1007/s11390-018-1841-4](https://doi.org/10.1007/s11390-018-1841-4).
- [23] M. Kim, B. Hilton, Z. Burks, and J. Reyes, "Integrating blockchain, smart contract-tokens, and IoT to design a food traceability solution," in *Proc. IEEE 9th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON)*, vol. 2018, Jan. 2019, pp. 335–340, doi: [10.1109/IEMCON.2018.8615007](https://doi.org/10.1109/IEMCON.2018.8615007).
- [24] S. K. Lo, X. Xu, M. Staples, and L. Yao, "Reliability analysis for blockchain oracles," *Comput. Electr. Eng.*, vol. 83, May 2020, Art. no. 106582, doi: [10.1016/j.compeleceng.2020.106582](https://doi.org/10.1016/j.compeleceng.2020.106582).
- [25] V. Thiruchelvam, A. S. Mughisha, M. Shahpasand, and M. Bamiah, "Blockchain-based technology in the coffee supply chain trade: Case of burundi coffee," *J. Telecommun., Electron. Comput. Eng.*, vol. 10, no. 3, p. 121–125, 2018.
- [26] G. Leduc, S. Kubler, and J.-P. Georges, "Innovative blockchain-based farming marketplace and smart contract performance evaluation," *J. Cleaner Prod.*, vol. 306, Jul. 2021, Art. no. 127055, doi: [10.1016/j.jclepro.2021.127055](https://doi.org/10.1016/j.jclepro.2021.127055).
- [27] H. A. Palacios-Cabrera, H. C. Menezes, B. T. Imanaka, F. Canepa, A. A. Teixeira, N. Carvalhaes, D. Santi, P. T. Z. Leme, K. Yotsuyanagi, and M. H. Taniwaki, "Effect of temperature and relative humidity during transportation on green coffee bean moisture content and ochratoxin a production," *J. Food Protection*, vol. 70, no. 1, pp. 164–171, Jan. 2007, doi: [10.4315/0362-028X-70.1.164](https://doi.org/10.4315/0362-028X-70.1.164).
- [28] E. K. Wati, F. Hidayanti, and A. Prasetya, "Design of temperature and humidity control on Arabica coffee storage," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 4, pp. 1076–1080, 2020, doi: [10.35940/ijitee.d1542.029420](https://doi.org/10.35940/ijitee.d1542.029420).
- [29] T. H. Pranto, A. A. Noman, A. Mahmud, and A. B. Haque, "Blockchain and smart contract for IoT enabled smart agriculture," *PeerJ Comput. Sci.*, vol. 7, pp. 1–29, Mar. 2021, doi: [10.7717/PEERJ-CS.407](https://doi.org/10.7717/PEERJ-CS.407).
- [30] C. Lacambra, D. Molloy, J. Lacambra, I. Leroux, L. Klossner, M. Talari, M. M. Cabrera, S. Persson, T. E. Downing, E. Downing, B. Smith, M. Abkowitz, L. A. Burnhill, and L. Johnson-Bell, "Factsheet resilience solutions for the maize sector in South Africa," Inter-Amer. Develop. Bank, Washington, DC, USA, Jun. 2020, doi: [10.18235/0002405](https://doi.org/10.18235/0002405).
- [31] D. Ongaro and J. Ousterhout, "In search of an understandable consensus algorithm," in *Proc. USENIX Annu. Tech. Conf. (USENIX ATC)*, 2019, pp. 305–319.
- [32] H. Treiblmaier, "Toward more rigorous blockchain research: Recommendations for writing blockchain case studies," *Frontiers Blockchain*, vol. 2, p. 3, May 2019, doi: [10.3389/FBLOC.2019.00003](https://doi.org/10.3389/FBLOC.2019.00003).
- [33] A. Sharma, D. Agrawal, F. M. Schuhknecht, and J. Dittrich, "Blurring the lines between blockchains and database systems: The case of hyperledger fabric," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2019, pp. 105–122, doi: [10.1145/3299869.3319883](https://doi.org/10.1145/3299869.3319883).



**CRISTIAN VALENCIA-PAYAN** received the bachelor's degree in physical engineering and the master's degree in telematics engineering from the Universidad del Cauca, Cauca, Popayán, Colombia, in 2012 and 2018, respectively, where he is currently pursuing the Ph.D. degree in telematics engineering. He is also currently in charge of the technological inclusion in the production processes of traditional sweets in the city of Popayán. His current research interests include computer vision, remote sensing, multisensor data fusion, blockchain technology, and the IoT sensors.



**JOSÉ FERNANDO GRASS-RAMÍREZ** received the master's degree in rural development from the Pontificia Universidad Javeriana, Bogotá, and the Ph.D. degree in agro-industrial economic sciences from the Universidad Autónoma Chapingo, Mexico. He is a Chemical Engineer, with an emphasis in food science and technology, with the Universidad Nacional de Colombia, Bogotá; and a specialist in production management with ICESI University, Cali. He has been a Full Professor with the Universidad del Cauca, since 2000, and has held positions such as the Dean of the Faculty of Agricultural Sciences, a Coordinator of the doctorate in agricultural and agro-industrial sciences, and a Coordinator of the master's degree in agricultural sciences and the Agro-Industrial Engineering Program. He is part of the Tull Research Group, which is classified in category A according to the Ministry of Science and Technology. He currently directs the Project Strengthening, the coconut production chain of the Pacific Coast of Cauca, Colombia, which is financed by the General Royalty Systems, and participates as the Technical Director of the Project Strategies, for the valorization of traditional sweets from Popayán. His research interests include the analysis of agro-industrial chains, value networks, networks analysis, and localized agrifood systems.



**GUSTAVO RAMIREZ-GONZALEZ** received the B.S. degree in electronic and telecommunications engineering and the M.S. degree in telematics engineering from the University of Cauca, Colombia, in 2001, and the Ph.D. degree in telematics engineering from the Universidad Carlos III de Madrid, Spain, in 2010. He is currently a Professor and a Researcher with the Department of Telematics, University of Cauca. He has participated in national and international projects in Colombia and Spain. He has published several research papers in reputed journals and served as a guest editor for several special issues at many journals. His research interests include image processing, secure communication, machine learning, and the IoT.



**JUAN CARLOS CORRALES** received the Dipl.-Ing. and master's degrees in telematics engineering from the University of Cauca, Colombia, in 1999 and 2004, respectively, and the Ph.D. degree in sciences, specialty in computer science, from Versailles Saint-Quentin-en-Yvelines University, France, in 2008. He is currently a Full Professor and leads the Telematics Engineering Group (GIT), University of Cauca. His research interests include machine learning and data analytics.