

Received December 12, 2021, accepted December 29, 2021, date of publication December 30, 2021, date of current version January 7, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3139829

Supply Chain Inventory Sharing Using Ethereum Blockchain and Smart Contracts

ILHAAM A. OMAR¹, RAJA JAYARAMAN¹, MAZIN S. DEBE², HAYA R. HASAN²,
KHALED SALAH², (Senior Member, IEEE), AND MOHAMMED OMAR¹

¹Department of Industrial and Systems Engineering, Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates

²Department of Electrical Engineering and Computer Science, Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates

Corresponding author: Raja Jayaraman (raja.jayaraman@ku.ac.ae)

This work was supported in part by the Khalifa University of Science and Technology under Award CIRA-2019-001, and in part by the Research Center for Digital Supply Chain and Operations Management under Grant RCII-2019-002-.

ABSTRACT Information sharing is essential for coordination and stakeholder integration and is crucial to achieving sustainable supply chain operations. Furthermore, improved information sharing assists retailers by reducing the time spent looking for alternative suppliers to meet the unexpected demand surge during pandemics. Recent advances in networking and distributed ledger technologies have enabled secure, improved information communication and broader connectivity. However, due to system interoperability, information quality, confidentiality, and trust, information cannot be exchanged seamlessly, resulting in the inefficiency of managing supply chain operations. In this paper, we propose a blockchain-based inventory sharing approach based on smart contracts using a private Ethereum network to link suppliers and retailers. Our approach combines blockchain technology with decentralized storage to increase transparency, trust, and security of supply chain transactions. We present a generalized mechanism for secure information sharing that includes comprehensive algorithms to capture supply chain stakeholder interactions that enhance trust among participating entities. The smart contract was developed and validated using the Remix IDE, and the code is made publicly available on Github. We analyze the solution for various security vulnerabilities and provide a detailed cost analysis for various stakeholder transactions in the supply chain. Our solution demonstrates that a blockchain-based approach reduces inefficiencies, is economical, commercially viable, and provides improved information connectivity among supply chain stakeholders in a trusted and secure way.

INDEX TERMS Inventory sharing, blockchain, Ethereum, transparency, supply chain disruption.

I. INTRODUCTION

The role of retailers in the supply chain has drastically changed since the 1980s. Previously, retailers held passive power positions where upstream suppliers would push products regardless of customer choice or retailer predicted demand. However, this practice has changed due to rapid developments in technology, competitive markets, and complex supply chains [1]. Retailers now have a critical role in the supply chain process, from production and distribution all the way to consumption, forecasting customer demand in real-time. The transforming role of retailers is affected by factors such as outsourcing non-core businesses, eliminating unnecessary inventory, and enhancing the role of information systems [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Mohamed Elhoseny¹.

Anand and Grover [2] argue that factors such as market instability, increased competition, globalization, and reduced product life cycles force retailers to be responsive and adaptive to external change. Furthermore, the development of supply chain management has been greatly influenced by advances in information systems and communications technology. Information sharing, in particular, has been significantly facilitated by these technical advancements, which has consequently resulted in strengthening the relationships among supply chain members. Nonetheless, organizations are being pushed to rethink their approach as global competition grows [3].

In fact, supply chain integration can be used both internally and externally by trading supply chain partners in a company. However, the true value of knowledge sharing within a supply chain can be observed when the benefits obtained outweigh the costs involved. These costs could include the

cost of computer systems as well as payments from vendors or consumers for supplying the data. Communication and management expenses can be included in the coordination costs. With new advancements in information technology, such as web applications and Enterprise Resource Planning (ERP), these costs may be drastically minimized [4].

There is a growing need for information sharing among supply chain trading partners as the economic climate is becoming increasingly volatile. Acquiring more knowledge about end-user needs is a key tool for reducing future demand uncertainty while eliminating excess inventory. Moreover, supply chain visibility not only allows managers to plan effectively but also allows them to respond appropriately to the right information. However, one of the most common barriers towards information sharing that is discussed in the literature is the lack of formal information systems. According to research published by SCM World, many businesses are hampered by the high investment costs and system deployment problems associated with formal information collaboration platforms [5].

Even if a company successfully manages to develop the necessary IT equipment to exchange information, issues related to trust and loyalty may prevent this development [6]. The information would not be shared with an organization that the managers do not trust, so mutual trust is another significant barrier to information exchange. Other barriers include fear of information theft and uncertainty about information quality. Therefore, in this paper, we propose using blockchain technology to connect suppliers and retailers in a trusted environment to encourage inventory sharing. This technology can enhance trust, improve efficiency, and coordination between partners, as well as enhance transparency in large supply chains involving complex transactions while simultaneously reducing disruptions. Our main contributions to this paper include:

- We discuss the importance of inventory sharing between suppliers and retailers and explain its benefits.
- We propose a private blockchain-based solution that uses Ethereum smart contracts and a decentralized storage system to facilitate inventory sharing among supply chain stakeholders.
- We present detailed algorithms depicting the interactions among supply chain stakeholders, including sequence diagrams that explain the proposed solution for inventory sharing.
- We validate the proposed solution by conducting detailed testing of various scenarios and system functionalities. We also provide cost and security analysis demonstrating the feasibility of our approach.

The rest of this paper is presented as follows: Section II highlights the benefits of inventory sharing and the challenges with existing sharing mechanisms. In section III, we discuss the existing use cases in the supply chain sector that have integrated blockchain technology into their systems. Section IV describes the proposed solution for inventory sharing using blockchain technology, while Section V

explains the implementation of the proposed solution along with the detailed algorithms employed in the smart contract. Section VI presents the results of testing and validation of the proposed solution, and Section VII discusses the cost and security analysis of the proposed solution. Finally, section VIII presents the conclusions and future work.

II. BACKGROUND

In this section, we explain the benefits of exchanging information between suppliers and retailers in a supply chain and the current challenges with inventory sharing mechanisms. We also highlight the importance of adopting blockchain technology in tackling these existing challenges.

A. BENEFITS OF INVENTORY SHARING

Numerous studies [7]–[9] discuss the types of information exchanged among supply chain partners. One of the most common types of data that supply chain participants want to exchange is inventory information [10]. Having up-to-date access to inventory status would help to reduce overall inventory levels in the supply chain. For example, consider two firms in a supply chain where a retail store buys goods from a supplier. If the supplier and retailer handle their inventories independently without sharing their inventory status details, there is a high possibility that they might end up with duplicate safety inventories or even stockouts at both locations. This demonstrates that exchanging information such as inventory levels, product specifications, safety stock levels, order information, and performance indicators such as distribution performance and product quality within the supply chain network is crucial for running a firm efficiently.

For instance, the Vendor-Managed Inventory (VMI) scheme is one of the most widely used systems for optimizing supply chain performance [11]. It requires supply chain members to collaborate and share inventory information in which the supplier makes the necessary replenishment decisions depending on the current inventory of the retailer and the specific terms and conditions decided upon by both parties. Hence, retailers are required to be transparent about both their own stocking point and the customer receiving point in order to make the right replenishment decisions.

However, the VMI scheme encourages one-way information flow from the retailer to the supplier, in which the supplier alone makes all the necessary decisions related to inventory. This scheme works well under normal conditions; however, if there is a surge in demand, such as a pandemic, it will become problematic. This is because there is an unexpected high demand that neither suppliers nor retailers anticipated. Hence, inventory sharing mechanisms that promote knowledge, decisions, and resources to be shared among suppliers result in the development of a sustainable supply chain. This is because it aids in the improvement of knowledge flow and delivery performance, the reduction of uncertainty, and the creation of a more profitable supply chain [12]. Integrating suppliers into one network helps in the attainment of substantial benefits for everyone.

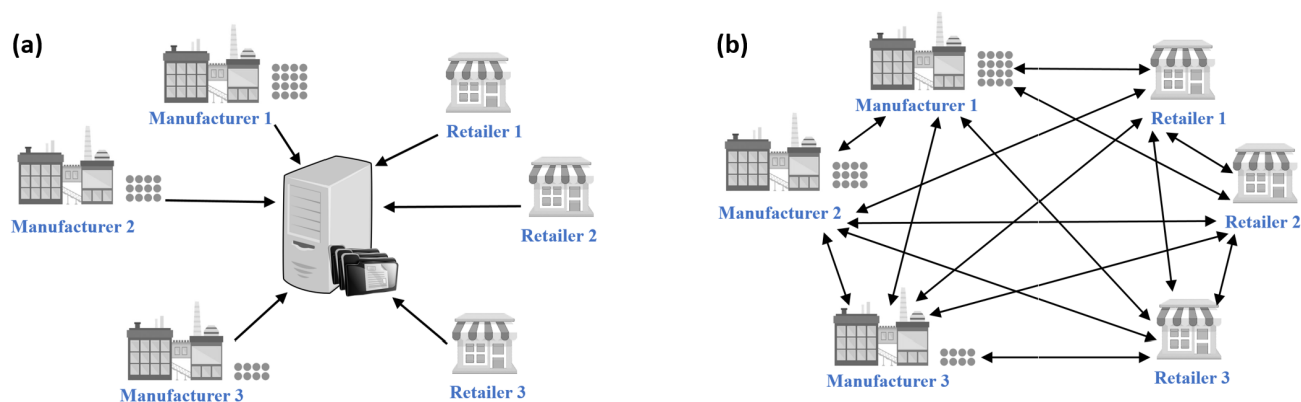


FIGURE 1. Overview of a (a) centralized supply chain inventory system and a (b) decentralized supply chain inventory system.

Additionally, businesses become more cautious when sharing information in order to alert current plans or devise potential operations by gathering additional information [10]. They may also use the information to fulfill current strategies and create new ones for the future, such as reducing inventory levels, lead times, or even expanding their operations. Allowing suppliers to have greater exposure will also result in cost savings, improved in-stock efficiency, increased revenue, and improved customer satisfaction. Furthermore, allowing information to be shared among different actors will result in a faster response and early detection of problems. Also, suppliers and retailers may come up with ideas to optimize their capacity usage as they improve their information-sharing mechanisms. It helps businesses to schedule and organize their operations and procedures more effectively. For example, Lee and Whang [13], argued that by exchanging information, businesses could be prepared for potential shortages. This could also result in cost savings in areas such as human resources and equipment.

B. CHALLENGES WITH EXISTING INVENTORY SHARING MECHANISMS

Companies may only trust each other if they believe that each stakeholder in the supply chain is interested in the profit and well-being of others within the same network. However, even if each stakeholder is guaranteed a profit, there would still be room for certain members to play a non-cooperative game. As a result, trust is required for businesses to share information. On the other hand, building this trusting relationship has never been easy for businesses, particularly in an unbalanced relationship. In fact, knowledge would not be shared with an organization that the managers do not trust, so mutual trust is a major barrier to information exchange.

Secondly, any actor in a retailer-supplier interface must work hard to earn the trust of the other entity. Furthermore, for any supply chain member, a constant and basic concern is how much information is adequate to share with partners in order to achieve expected profits without revealing sensitive information. As a result, businesses are unable to exchange confidential information on a centralized network for fear that

competitors will use it to compete with them [13]. According to Lotfi *et al.*, another major issue with information sharing is information privacy and security [10]. On the other hand, failing to collaborate with other manufacturers in a centralized network may result in an increase in inventory storage costs for suppliers and an increase in unsatisfied customers for retailers, unlike in a decentralized network, as shown in Fig. 1.

Moreover, successful cooperation is often hampered by a lack of reliable data on goods, pricing, competition, and purchases. These include inaccurate data collection, unclear modifications to decisions about what to share and what not to share, conflicting data distribution formats, in-store actions and decisions not reflected in the data, irregular frequency of data feeds, and a lack of omnichannel integration. Also, retailers and suppliers frequently lack clear internal metrics for managing and measuring the success of their digital partnership efforts. Traditional key performance indicators (KPIs) are too focused on revenue and sales, in which consumer centricity is not rigorously assessed, and each channel has its own KPIs that do not take into account the impact of other stakeholders in the network. The lack of data sharing and trusted data is often traced back to the use of divergent KPIs.

According to Lee and Whang, significant investments are required to enable information to be shared among various actors [13]. Many businesses are held back by the high investment costs and system deployment problems that come with formal collaboration partnerships. As a result, businesses can fail to adopt such an information system, in spite of recognizing it allows them to share information and therefore provide many benefits. Therefore, financial constraints also act as a significant obstacle to information sharing.

Despite the fact that IT infrastructure is no longer regarded as a major issue due to the rapid advancement of information technology, it still continues to impede information sharing. Implementing information sharing systems is strenuous, according to Lee and Whang, since participants can disagree over technical system requirements [13]. Also, different

organizations can use different types of applications, processes, data specifications, and so on, complicating information exchange. Another element of technical problems is a lack of practical preparation and expertise, especially for emerging technologies, which creates barriers to sharing information. Lastly, the fear of information theft and uncertainty about the accuracy of information increases their reluctance to exchange information.

C. ENHANCING INVENTORY SHARING USING BLOCKCHAIN TECHNOLOGY

When it comes to enhancing retailer results, the emphasis has traditionally been on their own operations [14]. However, the fact is that they must build and incorporate strategic initiatives that involve other stakeholders in the supply chain [15]. The way supply chain participants share information and organize their decisions, according to Chen, has a significant effect on the supply chain outcome [16]. To put it another way, the degree of integration among supply chain members decides the level of information exchange, which consequently affects supply chain performance.

However, reaching the desired level of performance in a more complicated market with longer and larger supply chains is getting even more complex. A possible reason for this is the lack of information sharing and data manipulation. The bullwhip effect is one of the most common examples of how information distortion can lead to poor supply chain efficiency [17]. Poirier and Bauer concluded that a successful supply chain is not limited to a single company, but to the network [18]. As the retailer is positioned at the end of the supply chain, it becomes essential that they perform well in order to have a positive impact on the whole chain. Under this situation, information sharing between a retailer and its supplier is crucial for the retailer to effectively handle critical operations, including resource utilization, inventory, customer satisfaction, and incurred expenses [10].

Companies have started to give their collaborators visibility by giving them access to business data by user accounts such as inventory. Users can now take advantage of a business information system by using software-as-a-service (SaaS) [19]. Information sharing and inter-organizational cooperation become more complex when customers and other organizations are involved. In fact, supply chain networks are nowadays dispersed, with suppliers scattered across several countries. Blockchain has been used in supply chain areas to improve transparency, authenticity, and trustworthiness [20]. The aim of this research is to study how incorporating blockchain technology into the supply chain will result in establishing a more transparent supply chain that encourages inventory sharing. Moreover, sharing sensitive information on a blockchain network would enhance visibility, improve coordination, enhance efficiency, and reduce disruptions.

We propose that supply chain members use Blockchain technology to ensure that the information they share is accurate and trustworthy. A blockchain is referred to as a form of a public ledger that contains a list of a chronological

sequence of time-stamped records secured by cryptography technology. The blocks are comprised of all of the transactions made by the participants. Computers can interact directly with one another on the blockchain, and each actor in the network is given a pair of private and public keys. Since each correspondence is signed with the private key of the sender, communication between the networks is safe. Each actor in the network would be provided with a wallet that can be used to certify the validity, invariability, and integrity of data through the use of the publicly available hash sum on the blockchain, while the original data can be stored off-chain using decentralized storage technologies. As a result, blockchain offers clarity and provides a consistent, secure, and single point of truth [21].

III. RELATED WORK

In this section, we discuss the relevant literature on the application of blockchain technology in the supply chain industry, in particular in relation to inventory management. Dujak and Sajter discussed the need for having an open-access network in supply chains [22]. They stated that having open access to information in the supply chain can result in advantages such as faster paperwork handling, reduced direct communications, and the provision of more information for the end-user.

In fact, IBM and Maersk have been designing freight tracking technologies, especially for containers, as well as applications for the digitalization of global trade [23]. To do so, they needed to form an open partnership with other participants, including Microsoft, DuPont, and others, to secure transactions and create a tamperproof registry based on open blockchain principles intended for use by the entire global shipping ecosystem. The preparation of paperwork and records for container shipping is expected to cost as much as the actual transportation [23]. They were able to ensure that all paperwork and transactions in the supply chain were accessible and transparent to all partners. This was done using a blockchain application that built a digitized document workflow that included details of all the participants' transactions and moved products, such as who, when, and where they were issued or moved.

Hence, establishing a blockchain platform reduces the need for direct contact on a domestic and international level, eliminates errors, delays, and other sources of waste, and ensures considerably smoother information transactions and, indirectly, faster inventory movements in the supply chain [22]. All information becomes available and decentralized, which reduces fraud and delays. The main advantages for ports, airports, ocean carriers, and intermodal transporters will be reliable and real-time knowledge about the disposition of shipments, allowing for end-to-end visibility of supply chain activities and more effective scheduling and preparing for their own operations. Furthermore, two of the largest ports in Europe, Rotterdam and Antwerp, also acknowledged the blockchain potential [24].

Blockchain has the capability to influence any sector in the supply chain, not just transportation and logistics. For instance, it can be used alongside the RFID system in the manufacturing supply chain to verify the origin of products, as both traceability and visibility will be enhanced [25]. Blockchain could also aid in demand forecasting as data stored in the blockchain would be immutable and reliable. Furthermore, this technology would help in fraud prevention as users would be required to verify their authenticity. This feature is particularly useful in the pharmaceutical supply chain and the luxury jewelry industry as it could aid in reducing counterfeit products [26], [27]. Lastly, using smart contracts in supply chains helps in transaction automation as they eliminate the need for third-party intervention [28]. A closer glance at current or emerging smart contract projects shows that supply chain management, Industry 4.0 solutions, and the Internet of Things are the most popular smart contract implementations in the market [29], [30].

In a supply chain, blockchain enables the integrated flow of information, capital, and materials that facilitates system restructuring. Gold State Foods is a major supplier of services to restaurants that uses blockchain to track, trace, and monitor its food supply. Moreover, The Chain Integration Pilot (CHIP) project was launched in 2018 by Nike, Macy's, and Kohl's. CHIP has successfully uploaded 223,036 serialized goods data points through a distributed ledger, enabling partners to share these data with each other [30]. Ozer *et al.* also examine the importance of information quality in information sharing [31]. In a military procurement context, Zaerens [32] proposes the use of smart contracts for monitoring the trading balance and information quality between participants.

Since blockchain research in supply chain management is still in its infancy, it is worth looking at potential implementations that might persuade supply chain managers to use this platform as a certification agent for mutual data and information. The COVID-19 pandemic, we conclude, has emphasized the need for increased supply chain visibility and productivity to enable inventory sharing in situations like inventory shortages or demand spikes. As a result, we suggest a solution designed exclusively for inventory sharing to improve the visibility of stakeholders across the supply chain and the productivity of business processes. In fact, Table 2 compares our proposed solution to the recent papers discussed in the supply chain sector. The following sections discuss the details of our proposed approach, as well as provide validation, cost, and security analysis.

IV. PROPOSED BLOCKCHAIN-BASED INVENTORY SHARING SOLUTION

We propose a private blockchain-based solution for facilitating information exchange, such as inventory between retailers and suppliers in a network. Our solution adopts blockchain technology to promote transparency, encourage trust, and enhance information security between the stakeholders in the supply chain. Moreover, our system develops smart

contracts and uses decentralized storage technology in the Ethereum network, as demonstrated in Fig. 2. This would allow registered retailers to track the inventory level in real-time for each supplier in the network.

A. STAKEHOLDERS AND COMPONENTS

We describe the roles of each stakeholder and component in the proposed system below.

- **Supplier:** The supplier researches the market, determines the necessary goods, and calculates the required amount to fulfill consumer demand while ensuring that the products meet regulatory requirements. Simultaneously, a supplier must be prepared to anticipate incidents that could result in commodity shortages, such as raw material delays during natural disasters or manufacturing disruptions during pandemics.
- **Retailer:** In a supply chain, a major responsibility of the retailer is to provide consumers with a variety of goods sourced from numerous producers, wholesalers, dealers, or suppliers. As a result, the retailer is the ultimate step in the supply chain, connecting vendors to consumers.
- **Ethereum Smart Contracts:** A code that is created to automatically execute functions based on the terms and conditions decided upon by network stakeholders. It works as a virtual agent that verifies transactions without the need for third-party intervention. Our proposed approach consists of four smart contracts, each of which is focused on a specific task. The registration contract focuses on registering all network members, while the inventory level contract specifies the amount of inventory left for each supplier as well as the product descriptions offered by suppliers. The order management contract deals with handling and managing orders, and the reputation contract ranks vendors based on their product quality, distribution efficiency, and honesty.
- **Decentralized storage technology:** Using a peer-to-peer distributed file system like Filecoin or Interplanetary File System (IPFS) makes it easier to link all nodes in a network to the same file system [33]. Authorized network users can use IPFS to store vast volumes of data and use blockchain transactions to store the permanent IPFS hashes, which are time-stamped and encrypted using cryptographic technologies. This is advantageous because it eliminates the need for consumers to store data on the chain. As a result, the combination of blockchain and distributed storage is advantageous since nodes in the network can only store indexed data, making it easier to find where data is stored.

B. METHODOLOGY

We initially identified the main stakeholders and components discussed in Section IV.A. The proposed system was built using this information as shown in Fig. 2. This system is then captured as a series of functions and events in the sequence diagram shown in Fig. 3 highlighting the interaction between

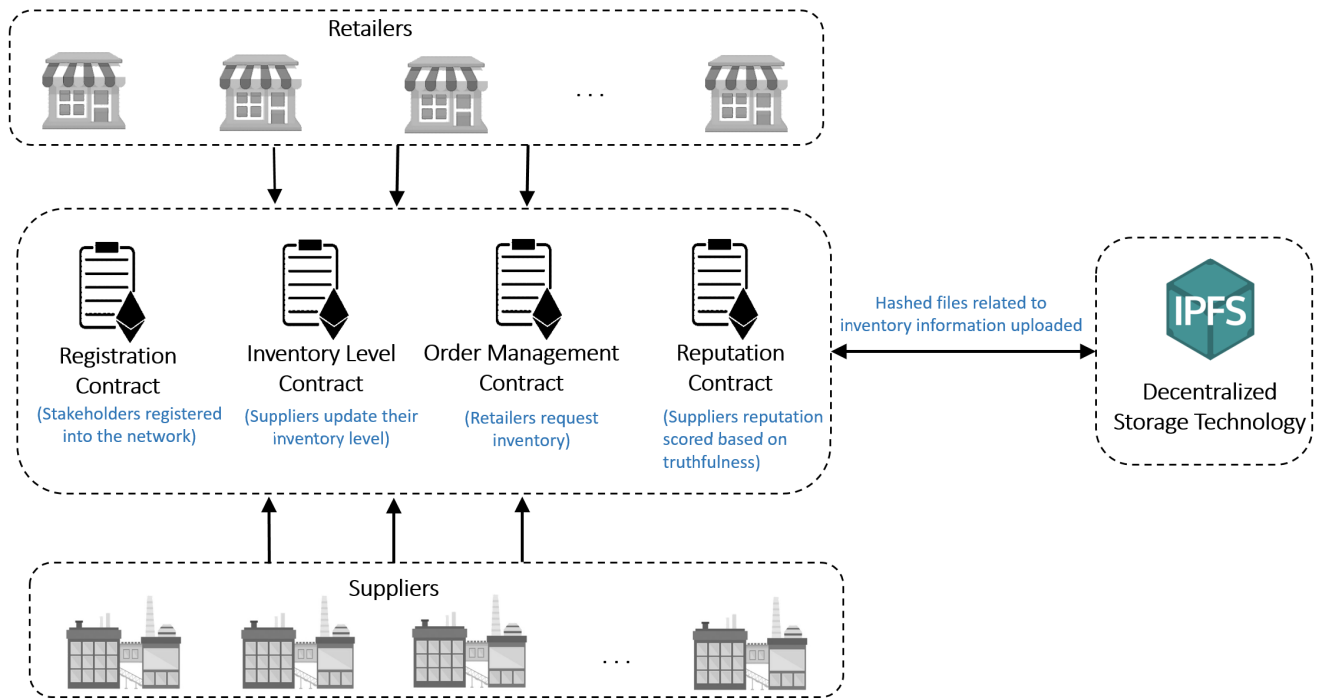


FIGURE 2. System overview of a blockchain-based inventory sharing mechanism using Ethereum smart contracts and decentralized storage system.

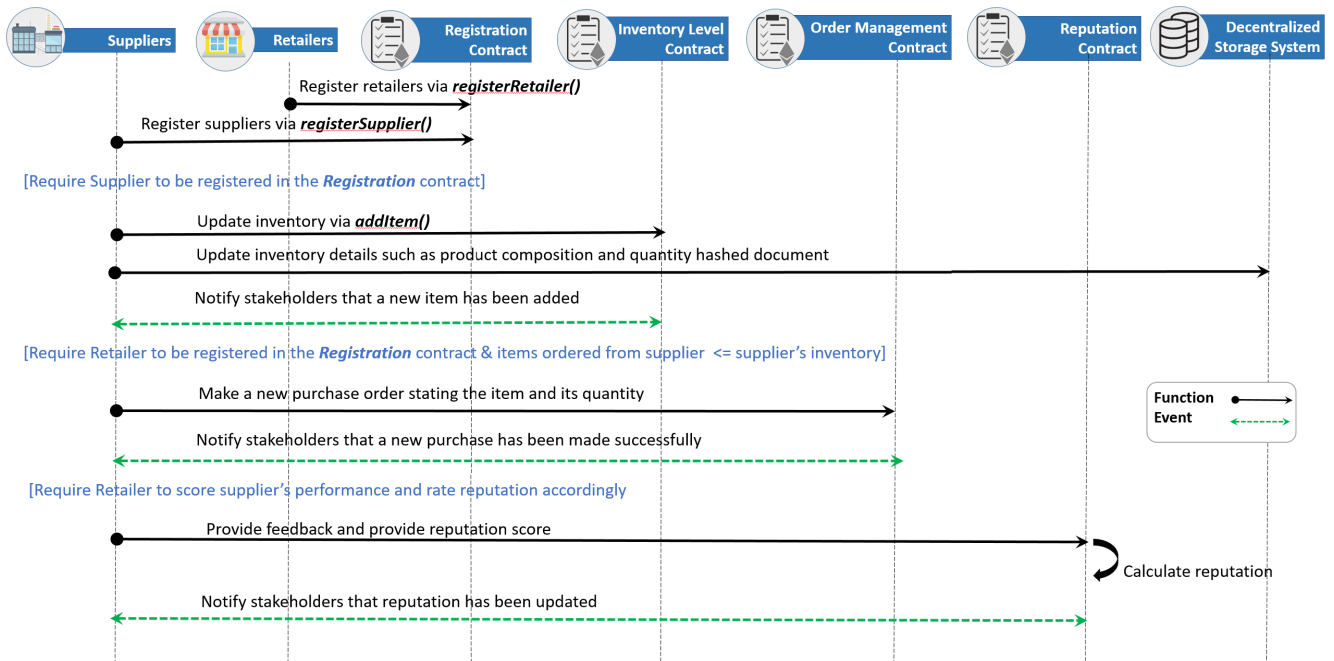


FIGURE 3. System overview of a blockchain-based inventory sharing mechanism using Ethereum smart contracts and decentralized storage system.

each stakeholder and contracts. The code is then designed and tested according to the proposed sequence diagram.

Initially, each stakeholder in the network is registered using their Ethereum address in the registration contract. Retailers and suppliers are registered by calling the functions `registerRetailer()` and `registerSupplier()` respectively. The system then allows the registered suppliers to upload their

inventory information, such as product numbers, quantities, descriptions, etc., by calling the `addItem()` function found in the inventory level contract. This triggers an event for all the stakeholders that a particular item has been uploaded successfully. The registered retailers use this information to make a new purchase order stating the product and quantity needed. Then an event is triggered depending on the status

of the order, alerting the stakeholders whether the order is accepted or rejected.

When an order is placed with a particular supplier, the number of items in the inventory gets deducted from the inventory level contract automatically. Thus, whenever retailers make a successful order, they are allowed to rate the performance of their suppliers using a reputation contract. The retailers call this contract and provide feedback, which automatically calculates the reputation score of the suppliers and simultaneously notifies all stakeholders that the reputation score has been updated. This allows retailers to work with trusted suppliers while keeping suppliers aware that their actions get recorded in an immutable distributed ledger.

C. PRIVACY, CONFIDENTIALITY AND AUTHORIZATION

Preserving the privacy of the stake holders is vital while successfully communicating and interacting using the blockchain network functions. Therefore, we have chosen a permissioned blockchain. Each type of permissioned blockchain has its way for maintaining privacy and confidentiality amongst its participants. For instance, Hyperledger Fabric is a permissioned network that is known for its plug and play design. Its versatile architecture relies on Membership Service Providers (MSPs) as well as certificate authorities to provide roles for the participants as well as identities and communication rights. Therefore, authorization is achieved through members allowed in after registering through the MSPs. Moreover, hyperledger fabric also provides channels which are ways for certain stake holders to subscribe in and communicate privately. Communication within a channel is transparent only to its participants. In addition, any information communicated within a channel can also be encrypted to achieve confidentiality [34]. On the other hand, other permissioned networks such as the java based Ethereum permissioned Besu network does not support permissioning itself. However, it relies on Ethereum-compatible wallet called EthSigner to manage the private keys of its participants. Furthermore, Orion is used to ensure a private communication is possible between a set of users [35]. There are many private blockchain based networks that can be used and our solution is easily adaptable to any of them. We have chosen a permissioned Ethereum network but our solution can also be implemented on Hyperledger Fabric and others.

Using a permissioned network allows the stake holders to have roles on the ledger and a private communication between certain entities based on the need. The communication is signed using the keys of the participants and encrypted if needed. Therefore, this ensures trust and enforces confidentiality as well as privacy. Only authorized entities can communicate in private channels or network subsets and groups.

The registered participating entities can execute functions in the smart contracts that they are authorized to only, based on their roles. Unregistered entities cannot take part in

executing any transaction. The transaction logs stored in the distributed ledger are accessible to the registered members only. Based on the type of the permissioned network, either channels or groups are created for private transactions taking place between a set of registered members. Hence, transaction logs of private transactions are accessible by members of the channel or group respectively. Therefore, each registered member can only execute their own transactions only based on their affiliated role and can only access logs that they are permitted to. Our solution leverages the intrinsic features of the blockchain as well as the permissioned features of the private blockchain.

D. REPUTATION

Using a reputation is important to elevate trust and transparency between the stake holders. It also serves as a rewarding mechanism for good and honest entities. Retailers can evaluate their experience with the suppliers by submitting a reputation score. Based on the performance of the supplier a score is awarded by the retailer. Moreover, suppliers can also evaluate their experience with a retailer once the transaction is completed. This helps in rewarding both retailers and suppliers and recalculating their scores after a new reputation score is submitted.

V. IMPLEMENTATION

This section presents the implementation details of the permissioned blockchain-based solution for promoting inventory sharing among retailers and suppliers in a network. The system overview presented previously was implemented on a test Ethereum blockchain. The Remix IDE was exploited to deploy smart contracts using the Solidity language.

There are four different types of smart contracts in our solution: registration smart contract, inventory level smart contract, order management smart contract, and reputation smart contract. These smart contracts are used for different purposes, as explained below.

- **Registration smart contract:** The owner of the system deploys this smart contract. The registration smart contract is only implemented once and is used to handle all stakeholder addresses. The Ethereum Address is a 20-byte identifier that is used to identify each stakeholder (EA). This smart contract includes both retailers and suppliers. In addition, it is used by other smart contracts to validate the identity of the entity through its EA. The registration smart contract registers members and only allows authenticated individuals with predefined roles from accessing the smart contract functions and on-chain transactions. Our modifiers restrict the registered members based on their roles.
- **Inventory level smart contract:** When a registered supplier's inventory level changes, this smart contract is updated. Each product is known by its EA, and once the contract is established, all of the product data is contained in it. The type of commodity sold by the seller, its name, the content and material hash file, the

CE certificate hash file, the overall quantity initially purchased, and the remaining quantity after distribution to different retailers are all included. Furthermore, this smart contract keeps track of each supplier's remaining inventory, making it easier for retailers to place orders accordingly. Every adjustment to each remaining quantity is immediately recorded and reflected on the blockchain. The registration smart contract fixed EA is also saved for comparison in order to verify the stakeholder identities.

- **Order management smart contract:** Registered retailers contact suppliers to place their orders for individual products and quantities. If the current supplier with whom the retailer is working does not have the quantity required, the retailer can place an order with another registered supplier in the network. These procedures are carried out by this smart contract, which declares new order placements. Before placing the order, the smart contract verifies the identities of the participants. Following that, all modifications to the orders, including order confirmation and execution, are registered in this smart contract. A special 32-byte number identifies each order here. This identifier is linked to the commodity ID, the quantity requested, the ordering entity address, and the order receiver address.
- **Reputation smart contract:** This smart contract is dedicated to providing trust in the integrity of the suppliers. Trusting the suppliers provides retailers with confidence in the system and the overall quality of service. The owner of the system deploys this contract, just like the registration smart contract. The retailers can access it to learn about the reputation score of the suppliers and to rate their interactions with them.

The proposed approach for inventory sharing comprises several smart contracts that contribute to the different processes in this solution. The pseudocodes for these are represented in the following algorithms. The purpose of these algorithms is to provide a brief summary of how the smart contract code operates. Algorithm 1 presents how the suppliers can add items to their inventory and increase their stock of existing items. Assuming the supplier is already registered in the Registration smart contract, the supplier deploys only once a different type of smart contract, called the Inventory smart contract. This is responsible for keeping track of the supplier inventory and modifying it. Naturally, only the Ethereum client that deploys such a smart contract can modify it. When the suppliers want to add an item, they enter its number, quantity, and price. However, if the item already exists, the supplier enters the item number and the quantity to be added. In addition, the price of an item can be adjusted through this smart contract. Finally, the newly updated quantity is sent out as an event for retailers to be updated.

All retailers can contact the Order Management smart contract to submit their purchase order requests as shown in algorithm 2. This smart contract manages the communication

Algorithm 1: Adding Items to the Supplier Inventory

Input: item Number, quantity to be added, item price
 1 Modifier: onlyOwner
 2 Supplier registers in the Registration smart contract.
 3 Supplier deploys the Inventory smart contract.
 4 Supplier enters item details to add in the Inventory smart contract.
 5 **if** *item number exists already* **then**
 6 | Increase the available quantity for that item.
 7 **else**
 8 | Set the quantity and price for the new item and add it to the smart contract.
 9 **end**
 10 **if** *price is changed* **then**
 11 | Adjust the price of the item to the new price.
 12 **end**
 13 Broadcast the quantity update by triggering the appropriate event.

between the retailers and the suppliers and their inventory smart contracts. The retailer provides the Ethereum address of the inventory smart contract it wishes to purchase from, along with the item number and the quantity ordered. The Order Management smart contract logs this request and checks whether the provided Ethereum address corresponds to a legitimate inventory smart contract. Consequently, the stock in this inventory is verified to have enough items for this order. The smart contract that manages this inventory then modifies the quantity available to account for the purchase order. Only the owner of the inventory smart contract is authorized to modify the quantity of the items available. The supplier is informed of this transaction through an event that includes the request details. The transaction is reverted at any point in the event of errors in the order, such as incorrect inventory addresses or unavailable items.

A separate smart contract is deployed to merely handle the reputation scores of suppliers. As shown in algorithm 3, the suppliers need to be pre-registered in this smart contract. During registration, the suppliers are given a preliminary reputation score that is adjusted based on their performance. Following a transaction with the supplier, a retailer can provide feedback about the quality of service of this supplier. Retailers are allowed to provide feedback about each supplier to keep the reputation score as fair as possible. The retailers provide their feedback about whether the interaction was satisfactory or not. According to this feedback, the reputation score is modified based on a constant set by the smart contract deployer. This constant determines how strict or lenient the smart contract is towards positive or negative feedback. In the case of positive feedback, the reputation score is multiplied by 0.95/16 and added to the original reputation score. In the case of negative feedback, the reputation score is multiplied by 0.95/24 and subtracted from the original reputation score.

Algorithm 2: Purchase Orders

Input: inventory address, item number, quantity ordered

- 1 Modifier: onlyRetailer
- 2 The retailer places a purchase order in the Order Management smart contract.
- 3 **if** *Inventory smart contract Ethereum address is valid* **then**
- 4 Get the available number of items for sale for this product from the Inventory smart contract.
- 5 **if** *quantity ordered is available in stock* **then**
- 6 Deduct the quantity ordered from the available stock in the supplier inventory smart contract.
- 7 Inform the supplier of the new purchase order via events.
- 8 **end**
- 9 **else**
- 10 Revert transaction.
- 11 **end**

VI. TESTING AND VALIDATION

This section is designated to provide testing details for the system's smart contracts. The developed code was validated to ensure it provides the required functionality and delivers the anticipated outcome. The Remix IDE was utilized to deploy and test the smart contract code. The code is easily accessible on GitHub¹. Furthermore, the Remix IDE provides a simulated test Ethereum network environment to write and verify smart contracts using the Solidity language. This validation is done by exploring the logs produced by the IDE for each transaction. These logs offer details about the transaction output, triggered events, gas cost, and exceptions thrown. In addition, the deployed code can be debugged to fix errors in the IDE itself. Remix provides enough details about the errors to make this process easier for developers. These include syntax checking, runtime error messages, as well as customizable error messages by the developer. Moreover, the development environment provides a good simulation of the mainnet by introducing gas limit restrictions on transactions that are also modifiable by the developer.

All the aforementioned smart contracts were deployed, each at a different Ethereum address on the test Ethereum network. The process of adding items to the inventory and submitting purchase orders was similar. Accounts (Ethereum addresses) were assigned to all relevant actors, and they were all registered in the Registration smart contract. Several accounts were for suppliers, and they were hence given each a SupplierInventory smart contract. For consistency, these smart contracts refer to the same Registration smart contract. The remainder of the accounts corresponded to retailers. The suppliers proceed to add their available items by using unique item numbers in addition to the available quantities. The retailers are updated about this via events that are triggered following any modification in the inventory. Hence, buyers

¹<https://github.com/MazenDB/InventorySharing/>

Algorithm 3: Updating Reputation Scores of Suppliers

Input: supplier Ethereum address, status of transaction

- 1 Validate that the Ethereum address of the given supplier was added to the Reputation smart contract.
- 2 **if** *retailer has already provided its feedback about this supplier* **then**
- 3 Revert transaction.
- 4 **else**
- 5 Accept feedback.
- 6 **if** *supplier provides acceptable service* **then**
- 7 The reputation score of the supplier is increased.
- 8 **else**
- 9 The reputation score of the supplier is decreased.
- 10 **end**
- 11 .
- 12 **end**

```
[ { "from": "0x64e8c7D246699db943917e8be1e812C2882aE322", "topic":
"0x4cfa5a83949d1abf4c5d97736486c80e9a80e754666891c6c84cabb258802c51", "event":
"ItemQuantityUpdated", "args": { "0": "886112723", "1": "50", "itemNum": "886112723",
"quantityAvailable": "50" } } ]
```

FIGURE 4. Event showing the updated quantity of a given item and its price.

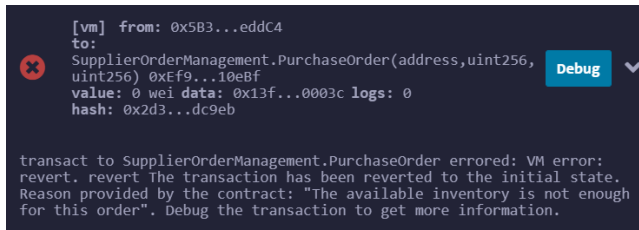
```
[ { "from": "0x64e8c7D246699db943917e8be1e812C2882aE322", "topic":
"0x4cfa5a83949d1abf4c5d97736486c80e9a80e754666891c6c84cabb258802c51", "event":
"ItemQuantityUpdated", "args": { "0": "886112723", "1": "40", "itemNum": "886112723",
"quantityAvailable": "40" } }, { "from": "0xEf9f1AC83dfb88f5590a621f4aE72C6EB10e8F",
"topic": "0x7b7f1b4f3b693b1c61972d7fd00568aa808829746ea46e79fa9a4f6cd5f9349b", "event":
"NewPurchaseOrder", "args": { "0": "0x64e8c7D246699db943917e8be1e812C2882aE322", "1":
"0x5B380a6a701c568545dCfcb83FcB875f56beddC4", "2": "886112723", "3": "10",
"inventoryAddress": "0x64e8c7D246699db943917e8be1e812C2882aE322", "retailer":
"0x5B380a6a701c568545dCfcb83FcB875f56beddC4", "itemNum": "886112723", "quantityOrdered":
"10" } } ]
```

FIGURE 5. The events triggered upon submitting a new purchase order by a retailer.

are able to view all available stock with their prices and availability. Fig.4 shows an event that is broadcast when a new item is added to the inventory, an existing item's stock has been increased, or the price of an item has been updated. This helps keep track of the latest updates regarding the supplier inventory.

Retailers can submit their purchase orders through the Supplier Order Management smart contract. Submitting a purchase order entails changing the available quantity of the ordered item. As shown in Fig.5, the available quantity of the stock is updated in the supplier inventory smart contract. In addition, another event is issued announcing the new purchase order and the requested order, including the item number, the order quantity, and the relevant inventory address. It is to be noted that all of these events are triggered only after validating the order and the corresponding inventory. If the retailer requests a quantity that exceeds the availability of the item, the order is rejected by throwing an exception, as can be seen in Fig.6.

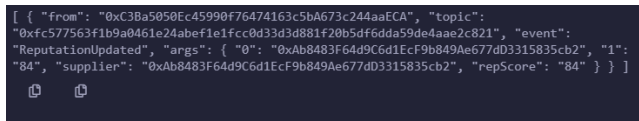
After the transaction is completed, the retailers have the option to evaluate the quality of service provided by the suppliers. This is done to give more priority to suppliers that are performing better than others. This is done through the



```
[vm] from: 0x5B3...eddc4
to:
SupplierOrderManagement.PurchaseOrder(address,uint256,
uint256) 0xEf9...10eBf
value: 0 wei data: 0x13f...0003c logs: 0
hash: 0x2d3...dc9eb

transact to SupplierOrderManagement.PurchaseOrder errored: VM error:
revert. revert The transaction has been reverted to the initial state.
Reason provided by the contract: "The available inventory is not enough
for this order". Debug the transaction to get more information.
```

FIGURE 6. Error message due to the quantity ordered exceeding the stock availability.



```
{ "from": "0xC30a5950Ec45990f76474163c5bA673c244aaEca", "topic":
"0xfc577563f1b9a0461e24abef1e1fcc0d33d3d881f20b5df6dda59de4aae2c821", "event":
"ReputationUpdated", "args": { "0": "0xAb8483f64d9c6d1EcF9b849Ae677d03315835cb2", "1":
"84", "supplier": "0xAb8483f64d9c6d1EcF9b849Ae677d03315835cb2", "repScore": "84" } }
```

FIGURE 7. Updating the reputation score of a supplier.

Reputation smart contract. The retailer gives its feedback, which is reflected in the reputation score of the supplier, as shown in Fig.7. In this case, the retailer reported a positive transaction that resulted in increasing the reputation score of the supplier. As mentioned previously, the reputation scores of the suppliers are capped at 100 at all times.

VII. DISCUSSION

In our paper, we implement a blockchain-based solution for inventory sharing between retailers and suppliers. Our proposed solution captures the main interactions that take place between these stakeholders. We will discuss the cost and security analysis of the proposed system.

A. COST ANALYSIS

Calling functions in Ethereum smart contracts has a cost measured in units of gas. The cost of a function call is contingent on the function itself and the gas price set by the caller. First, the function being called costs more with more complex computations and a higher number of inputs and outputs. Therefore, developers tend to avoid expensive operations such as loops to reduce the cost of operation. These determine the gas cost, which is a deterministic value. However, the Ethereum client has the freedom to select its own gas price, which is a mapping between gas and its cost in Ethers. The gas price is essentially the fee collected by miners on the blockchain network. Miners tend to pick transactions with higher gas prices and include them in their blocks before transactions with lower gas prices. Hence, a higher gas price would ensure a transaction is confirmed faster. Remix IDE provides a fairly accurate estimation of the transaction and execution gas cost for each transaction. These are mentioned in the logs produced by the IDE by computing them after the function call. This section aims to provide an estimation of the cost of operation of our system by showcasing these costs in terms of gas as well as converting them to US dollars. We chose a fiat currency for a better presentation of the transaction costs. These conversions are endorsed by the

Ethereum Gas Station that converts readings from the IDE to USDs for any arbitrary gas price provided [36].

Since the gas price is not a fixed value, the Ethereum Gas Station provides an approximation of the gas price by analyzing recent blocks. From those, it can deduce the gas prices for slow and cheap transactions, average transactions, and fast transactions. The gas costs for our function calls and their corresponding costs in USD are shown in Table 1. The gas prices assumed are 36, 44, and 55 Gwei, respectively. Gas prices fluctuate and will definitely change from the time of writing this paper. A more reasonable conversion rate from Ethers to USD is from May 2020, which is used in this cost analysis. Therefore, it can be seen that our solution is beneficial considering its cost. The execution cost of any function does not exceed \$0.45 for a slow transaction, \$0.55 for an average transaction, and \$0.75 for a fast transaction.

B. SECURITY ANALYSIS

In this section, we discuss the key properties of the proposed blockchain inventory sharing solution by addressing major security and privacy concerns related to trust, data integrity, availability, non-repudiation, and vulnerability to cyberattacks.

- **Confidentiality:** In terms of blockchain core functionality, pseudonyms are used to secure participant identities while also protecting their privacy. As valid transactions are time-stamped and stored in the blockchain, this encourages accountability so all players in the supply chain can be aware of what is going on at any given time. Furthermore, supply chain visibility is critical because it allows you to see how much inventory each registered supplier has. This is useful in situations where demand surges unexpectedly, such as pandemics, so retailers will know which suppliers have the appropriate goods in stock. Additionally, under the currently proposed system, all retailers and vendors are referred to by their EAs, ensuring that their names remain protected.
- **Data Integrity:** In a blockchain-based structure, integrity ensures that data stored in the network is accurate and that no one in the network is able to change or delete it. Furthermore, data integrity is protected when cryptographic protocols are used to encrypt valid transactions in a block. This is important since any order made by a retailer in the proposed system is deposited in the ledger as a valid contract, which becomes immutable and cannot be canceled or reversed. Simultaneously, the inventory levels of the suppliers will be reduced in real-time.
- **Availability:** This relates to the ability of the blockchain system to survive attacks and stay operational even when malicious code, such as a denial of service (DoS) attack, is present. This is attainable due to the hierarchical structure of blockchain technologies. As a result, even though certain nodes in the blockchain network are attacked, all functionalities such as ordering products,

TABLE 1. Gas cost of Ethereum functions in USD.

Method name	Transaction gas cost	Execution gas cost	Slow execution (USD)	Avg. execution (USD)	Fast execution (USD)
registerRetailer	45393	24121	0.21156	0.25857	0.34083
registerSupplier	45414	24142	0.21175	0.25879	0.34113
addItem	66272	44424	0.38964	0.47621	0.62772
deductItem	30998	9342	0.08194	0.10014	0.13200
PurchaseOrder	53521	30521	0.26769	0.32717	0.43127
addSupplier	44779	22099	0.19382	0.23689	0.31226
feedback	73846	50974	0.44709	0.54643	0.72028
calculateRep	57367	34687	0.30423	0.37183	0.49014

updating inventory information, and selling products would still be functional.

- **Non-repudiation:** Users that have been registered in the blockchain network cannot deny sending or accepting a particular transaction. This is advantageous because the supply chain network suppliers and retailers cannot deny that orders were placed improperly or that the inventory level is not reflected accurately.
- **Vulnerability to cyberattacks:** Cryptographic technology is used to digitally encrypt and secure transactions on the Ethereum network. As a result, any effort to tamper with any transfers will require knowledge of the private key. Furthermore, the attacker is unable to target a specific block since all blocks are cryptographically linked together. Furthermore, cyber-attacks like Man-In-The-Middle (MITM) and distributed denial-of-service (DDoS) are becoming almost impossible.

C. GENERALIZATION

Our suggested approach is intended to be generic enough to satisfy the demands and expectations of shared network providers and retailers. It can be used in any supply chain industry, whether it is agriculture or commercial. The deployment of Ethereum smart contracts will need to be made public so that customers of the supply chain network can see them and get warnings during unexpected demand spikes like pandemics. As a result, only minor modifications to the smart contract source code will be required to ensure that the entry data fields meet the specifications for the target product.

However, many cooperation projects and efforts will not progress past the proof-of-concept level until the C-suite commits to long-term digital collaboration. Leaders from both sides should cooperate on benefit-sharing models, invest in the necessary technological infrastructure and platforms, be able to develop and improve their operational models, motivate middle management capabilities, and be bold in testing out and scaling new and promising digital projects.

D. QUALITATIVE COMPARISON

Table 2 compares the work described in this research to other relevant solutions. This comparison is made based on the features developed by the authors of these papers in order to prove the advantage of our proposed approach for supply chain inventory sharing over other approaches. All of the mentioned work was aimed at developing blockchain-based

TABLE 2. Comparison with state-of-the-art.

Features	Guggenberger et al. [37]	IBM & Maersk [23]	Wang et al. [38]	Casino et al. [39]	our work
Inventory sharing	✓	✓	✓	✓	✓
Blockchain-based	✓	✓	✓	✓	✓
Smart contracts	✗	✓	✓	✓	✓
Decentralized storage	✗	✗	✗	✓	✓
DApps	✗	✗	✗	✗	✓
Publicly available	✓	✓	✓	✓	✓

solutions for inventory sharing. Some of the solutions lack support for decentralized storage, such as [23], [37], [38]. Moreover, it was found that none of those solutions offered DApp support. In contrast, our proposed solution provides a decentralized approach that orchestrates inventory sharing amongst supply chain actors, utilizing the Ethereum blockchain and its smart contracts, along with a decentralized storage system. Adding a Dapp to our solution would ease the communication between retailers and suppliers while simultaneously encouraging inventory sharing among the stakeholders. Secondly, DApps are immune to single points of failure, which means that inventory information will be readily accessible for all participants as the system no longer relies on a single entity, such as in centralized applications. Moreover, participants in the network would not be required to rely on external influencers such as government corporations or developers to facilitate their transactions.

VIII. CONCLUSION

This paper addressed the importance of inventory sharing among various stakeholders in a supply chain. We have discussed the relevance of information sharing to running an effective and profitable supply chain operation. Our proposed solution combines blockchain and decentralized storage technologies to improve trust, efficiency, and transparency in complex supply chains. Additionally, it helps strengthen the coordination between partners while reducing unnecessary delays. The proposed solution can be adapted to various product categories in the supply chain where the system architecture, algorithms, sequence diagrams, and

testing scenarios can be suitably customized. Our solution permits only registered stakeholders to communicate with the smart contract, maintaining the integrity of transactions, stakeholder trust, and accountability. The smart contract code is publicly available on GitHub for the benefit of research and practice communities. We have discussed the security of the proposed solution in terms of data integrity, non-repudiation, etc. We have analyzed the various transaction costs involved for each stakeholder. Our work demonstrates that blockchain-based information sharing solutions reduce inefficiencies, are economical, commercially viable, and provide improved information connectivity among supply chain stakeholders in a trusted and secure way. We propose developing decentralized applications to facilitate full automation of other relevant processes affecting all supply chain stakeholders as potential future work. We also acknowledge that as blockchain is still in its infancy, problems such as scalability, governance, and energy consumption still remain as open challenges to be addressed in the future.

REFERENCES

- [1] L. Sparks, "Supply chain management and retailing," in *Supply Chain Forum: Int. J.*, vol. 11, no. 4, pp. 4–12, 2010.
- [2] N. Anand and N. Grover, "Measuring retail supply chain performance," *Benchmarking: Int. J.*, vol. 22, no. 1, pp. 135–166, Feb. 2015.
- [3] B. B. Flynn, B. Huo, and X. Zhao, "The impact of supply chain integration on performance: A contingency and configuration approach," *J. Oper. Manage.*, vol. 28, no. 1, pp. 58–71, 2010.
- [4] J. Li, M. J. Shaw, R. T. Sikora, G. W. Tan, and R. Yang, "The effects of information sharing strategies on supply chain performance," *College Commerce Bus. Admin., Univ. Illinois at Urbana-Champaign, Champaign, IL, USA*, Sep. 2002. [Online]. Available: http://citebm.cba.uiuc.edu/B2Bresearch/ieeem_em.pdf
- [5] M. M. Ali, M. Z. Babai, J. E. Boylan, and A. A. Syntetos, "Supply chain forecasting when information is not shared," *Eur. J. Oper. Res.*, vol. 260, no. 3, pp. 984–994, Aug. 2017.
- [6] H. Mendelson, "Organizational architecture and success in the information technology industry," *Manage. Sci.*, vol. 46, no. 4, pp. 513–529, Apr. 2000.
- [7] X. Liu and Y. Sun, "Information flow management of vendor-managed inventory system in automobile parts inbound logistics based on Internet of Things," *J. Softw.*, vol. 6, no. 7, pp. 1374–1380, Jul. 2011.
- [8] R. Croson and K. Donohue, "Behavioral causes of the bullwhip effect and the observed value of inventory information," *Manage. Sci.*, vol. 52, no. 3, pp. 323–336, Mar. 2006.
- [9] N. Ebrahim-Khanjari, W. Hopp, and S. M. R. Iravani, "Trust and information sharing in supply chains," *Prod. Operations Manage.*, vol. 21, no. 3, pp. 444–464, May 2012.
- [10] Z. Lotfi, M. Mukhtar, S. Sahran, and A. T. Zadeh, "Information sharing in supply chain management," *Proc. Technol.*, vol. 11, pp. 298–304, Jan. 2013.
- [11] I. A. Omar, R. Jayaraman, K. Salah, M. Debe, and M. Omar, "Enhancing vendor managed inventory supply chain operations using blockchain smart contracts," *IEEE Access*, vol. 8, pp. 182704–182719, 2020.
- [12] S. M. Wagner, "Intensity and managerial scope of supplier integration," *J. Supply Chain Manage.*, vol. 39, no. 4, pp. 4–15, Sep. 2003.
- [13] H. L. Lee and S. Whang, "E-business and supply chain integration," in *The Practice of Supply Chain Management: Where Theory and Application Converge*. Boston, MA, USA: Springer, 2004, pp. 123–138.
- [14] P. M. Dunne, R. F. Lusch, and J. R. Carver, *Retailing*. Boston, MA, USA: Cengage Learning, 2013.
- [15] I. Plazibat and S. Brajevic, "Supply chain management in retail industry," *Bus. Logistics Mod. Manage.*, vol. 9, pp. 133–140, 2009.
- [16] F. Chen, "Information sharing and supply chain coordination," in *Handbooks in Operations Research and Management Science*, vol. 11. Amsterdam, The Netherlands: Elsevier, 2003, pp. 341–421.
- [17] P. Fiala, "Information sharing in supply chains," *Omega*, vol. 33, no. 5, pp. 419–423, 2005.
- [18] C. C. Poirier, M. J. Bauer, and M. J. Bauer, *E-Supply Chain: Using the Internet to Revolutionize Your Business: How Market Leaders Focus Their Entire Organization on Driving Value to Customers*. San Francisco, CA, USA: Berrett-Koehler, 2000.
- [19] F. Longo, L. Nicoletti, A. Padovano, G. d'Atri, and M. Forte, "Blockchain-enabled supply chain: An experimental study," *Comput. Ind. Eng.*, vol. 136, pp. 57–69, Oct. 2019.
- [20] S. Laaper, J. Fitzgerald, E. Quasney, W. Yeh, and M. Basir, "Using blockchain to drive supply chain innovation," in *Proc. Digit. Supply Chain Manag. Logist. Hambg. Int. Conf. Logist.*, vol. 1, Dec. 2017, p. 2013.
- [21] R. Azzi, R. K. Chamoun, and M. Sokhn, "The power of a blockchain-based supply chain," *Comput. Ind. Eng.*, vol. 135, pp. 582–592, Sep. 2019.
- [22] D. Dujak and D. Sajter, "Blockchain applications in supply chain," in *SMART Supply Network*. Cham, Switzerland: Springer, 2019, pp. 21–46.
- [23] M. Linnet and K. Mosquera, "Maersk and IBM to form joint venture applying blockchain to improve global trade and digitalize supply chains," Jun. 2018. [Online]. Available: <https://newsroom.ibm.com/2018-01-16-Maersk-and-IBM-to-Form-Joint-Venture-Applying-Blockchain-to-Improve-Global-Trade-and-Digitize-Supply-Chains>
- [24] I. J. Orji, S. Kusi-Sarpong, S. Huang, and D. Vazquez-Brust, "Evaluating the factors that influence blockchain adoption in the freight logistics industry," *Transp. Res. E: Logistics Transp. Rev.*, vol. 141, Sep. 2020, Art. no. 102025.
- [25] S. A. Abeyratne and R. P. Monfared, "Blockchain ready manufacturing supply chain using distributed ledger," *Int. J. Res. Eng. Technol.*, vol. 5, no. 9, pp. 1–10, 2016.
- [26] I. Haq and O. Muselemu, "Blockchain technology in pharmaceutical industry to prevent counterfeit drugs," *Int. J. Comput. Appl.*, vol. 180, no. 25, pp. 8–12, Mar. 2018.
- [27] L. E. Cartier, S. H. Ali, and M. S. Krzemnicki, "Blockchain, chain of custody and trace elements: An overview of tracking and traceability opportunities in the gem industry," *J. Gemmol.*, vol. 36, no. 3, pp. 212–227, 2018.
- [28] S. E. Chang, Y.-C. Chen, and M.-F. Lu, "Supply chain re-engineering using blockchain technology: A case of smart contract based tracking process," *Technol. Forecasting Social Change*, vol. 144, pp. 1–11, Jul. 2019.
- [29] G. Prause, "Smart contracts for smart supply chains," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 2501–2506, 2019.
- [30] Z. Wang, Z. Zheng, W. Jiang, and S. Tang, "Blockchain-enabled data sharing in supply chains: Model, operationalization, and tutorial," *Prod. Oper. Manage.*, vol. 30, no. 7, pp. 1965–1985, 2021.
- [31] Ö. Özer, Y. Zheng, and K.-Y. Chen, "Trust in forecast information sharing," *Manage. Sci.*, vol. 57, no. 6, pp. 1111–1137, Jun. 2011.
- [32] K. Zaearens, "Concept for controlled business critical information sharing using smart contracts," in *Proc. 2nd Cyber Secur. Netw. Conf. (CSNet)*, Oct. 2018, pp. 1–8.
- [33] J. Benet, "IPFS—content addressed, versioned, P2P file system," 2014, *arXiv:1407.3561*.
- [34] C. Cachin, "Architecture of the hyperledger blockchain fabric," in *Proc. Workshop Distrib. Cryptocurrencies Consensus Ledgers*, vol. 310, no. 4. Chicago, IL, USA, 2016, pp. 1–4.
- [35] H. R. Hasan, K. Salah, R. Jayaraman, I. Yaqoob, and M. Omar, "Blockchain architectures for physical internet: A vision, features, requirements, and applications," *IEEE Netw.*, vol. 35, no. 2, pp. 174–181, Mar. 2021.
- [36] *ETH Gas Station*. Accessed: Apr. 14, 2021. [Online]. Available: <https://ethgasstation.info/>
- [37] T. Guggenberger, A. Schweizer, and N. Urbach, "Improving interorganizational information sharing for vendor managed inventory: Toward a decentralized information hub using blockchain technology," *IEEE Trans. Eng. Manag.*, vol. 67, no. 4, pp. 1074–1085, Nov. 2020.
- [38] Z. Wang, T. Wang, H. Hu, J. Gong, X. Ren, and Q. Xiao, "Blockchain-based framework for improving supply chain traceability and information sharing in precast construction," *Autom. Construct.*, vol. 111, Mar. 2020, Art. no. 103063. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0926580519305059>
- [39] F. Casino, T. K. Dasaklis, and C. Patsakis, "Enhanced vendor-managed inventory through blockchain," in *Proc. 4th South-East Eur. Design Autom., Comput. Eng., Comput. Netw. Social Media Conf. (SEDA-CECNM)*, Sep. 2019, pp. 1–8.

• • •