


## Article

# agroString: Visibility and Provenance through a Private Blockchain Platform for Agricultural Dispense towards Consumers

Sukrutha L. T. Vangipuram <sup>1,†</sup>, Saraju P. Mohanty <sup>1,\*,†</sup>, Elias Kougianos <sup>2,‡</sup>  and Chittaranjan Ray <sup>3,‡</sup><sup>1</sup> Department of Computer Science and Engineering, University of North Texas, Denton, TX 76203, USA<sup>2</sup> Department of Electrical Engineering, University of North Texas, Denton, TX 76203, USA<sup>3</sup> Department of Civil and Environmental Engineering, University of Nebraska-Lincoln, Lincoln, NE 68588, USA

\* Correspondence: saraju.mohanty@unt.edu

† These authors contributed equally to this work.

‡ These authors contributed equally to this work.

**Abstract:** It is a known fact that large quantities of farm and meat products rot and are wasted if correct actions are not taken, which may lead to serious health issues if consumed. There is no proper system for tracking and communicating the status of the goods to their respective stakeholders in a secure way. Consumers have every right to know the quality of the products they consume. Using monitoring tools, such as the Internet of Agricultural Things (IoAT), and modern data protection techniques for storing and sharing, will help mitigate data integrity issues during the transmission of sensor records, increasing the data quality. The visibility state at the customer end is also improved, and they are aware of the agricultural product's conditions throughout the real-time distribution process. In this paper, we developed and implemented a CorDapp application to manage the data for the supply chain, called “agroString”. We collected the temperature and humidity data using IoAT-Edge devices and various datasets from multiple sources. We then sent those readings to the CorDapp agroString and successfully shared them among the relevant parties. With the help of a Corda private blockchain, we attempted to increase data integrity, trust, visibility, provenance, and quality at each logistic step, while decreasing blockchain and central system limitations.

**Keywords:** smart agriculture; Internet of Agricultural Things (IoAT); Agriculture Cyber-Physical System (A-CPS); private blockchain; CorDapp



**Citation:** Vangipuram, S.L.T.; Mohanty, S.P.; Kougianos, E.; Ray, C. agroString: Visibility and Provenance through a Private Blockchain Platform for Agricultural Dispense towards Consumers. *Sensors* **2022**, *22*, 8227. <https://doi.org/10.3390/s22218227>

Academic Editors: Himanshu Thapliyal and Akhilesh Tyagi

Received: 3 October 2022

Accepted: 24 October 2022

Published: 27 October 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

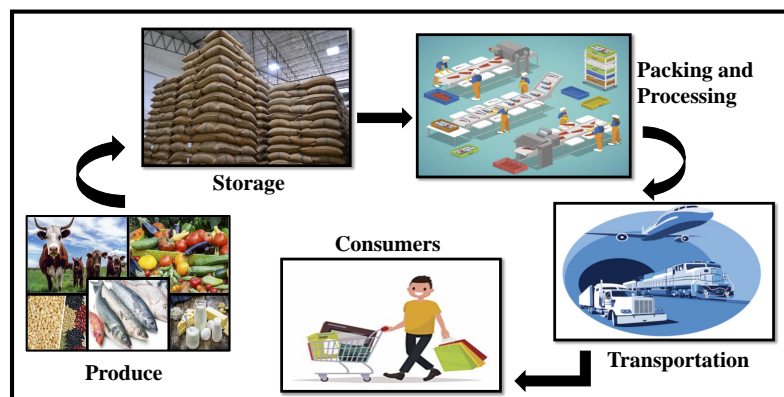


**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

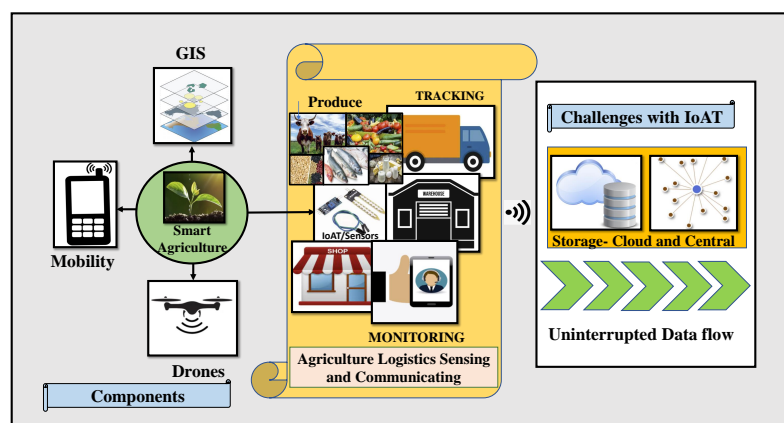
Agriculture plays a vital role in food production and is one of the primary sources of the farmers' daily livelihood. Food is the means of nutrition for the world population and is essential for increasing a country's economic status. Agriculture is comprised of farming crops, livestock, poultry, beekeeping, forestry, and sericulture. With the population projected to grow approximately from 6.9 billion people to 9.3 billion by 2050 [1], the demand for food production is expected to increase at a high rate. According to the U.S. Department of Agriculture, 40% of the produce is wasted and lost from the food supply every year [2] due to different factors, out of which distribution of farm produce contributes the highest percentage. As a result, more robust technological solutions are required to secure agricultural produce and transport fresh goods to the consumers, reducing wastage and providing trust in the process through visibility and provenance. The different trading options initiated, and transports instigated, have increased the accessibility of agricultural produce to even remote locations and long distances. Although shipping facilities have taken key roles in food supplies, factors like trust, quality, and time delivery impact the

agricultural supply chain, consequently increasing the cost of the product for the consumers directly [3]. Figure 1 shows different stages in agricultural produce distribution from producer to the end consumers.



**Figure 1.** Stages in agricultural product distribution towards the consumers.

Smart agriculture is one of the application fields of intelligent systems whose goal is to provide solutions for business processes using the IoAT (Internet of Agricultural Things). Advanced industries are using the IoAT and web-based solutions for incorporating modern techniques for data collection and the distribution of agricultural produce, along with implementing automation with security. Still, these approaches have limitations when bringing heterogeneous data into a unified system, raising privacy challenges and futile access control mechanisms [4]. Smart “things” employing a single point database system to store the statistics are having latency problems and Internet discontinuity issues for data flow and face possible attacks on the useful information [5]. Figure 2 demonstrates the components utilized in Smart Agriculture and some of the challenges of the IoAT in the agricultural product supply flow [4].



**Figure 2.** Components of smart agriculture and challenges of the IoAT.

Technologies like the IoAT, AI/ML, Robotics Systems, and the blockchain play an essential role in making agriculture smart. Some of the relevant research works in smart agriculture include studying the health of the crops for disease detection [6], monitoring the growth of crops [7], crop damage estimation-eCrop, aerial vehicles for detecting wild-fires [8], livestock tracking [9], and distributed ledger technologies for securing data sent in real-time [10,11].

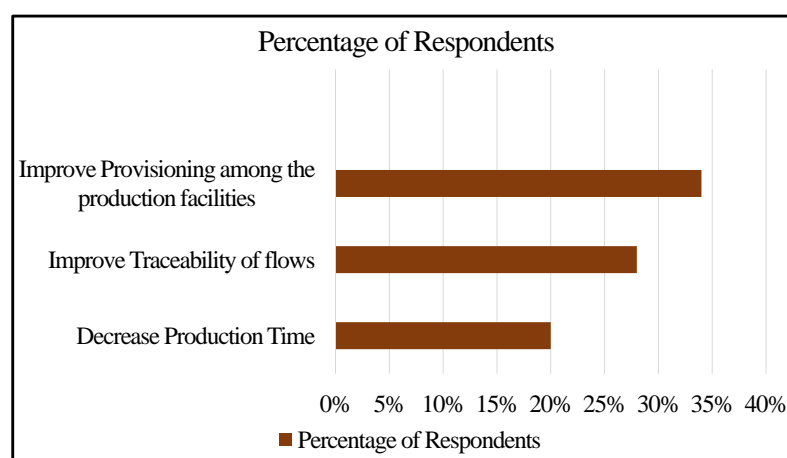
A blockchain is a distributed ledger that works on the principle of decentralization, allowing multiple parties to have a consistent view of the data transactions without a single authority or security embedded around it through cryptographic calculations. Participating nodes with copies of the written data can vote and agree to a single decision and make

the information valid by employing a consensus mechanism. Writing data onto the public blockchain can require rewards and computing power, consequently limiting the volume of data transfers and resulting in higher energy consumption. From the public domain cryptocurrencies, such as Bitcoin and Ethereum, the blockchain has evolved to private Enterprise Blockchain (EBC), which is more beneficial for industrial data exchange among relevant parties by mutual agreement.

The following is the order we present the current paper: Section 2 elaborates on some of the main issues that are present in the agricultural product distribution. Section 3 discusses the novelty through the usage of CorDapp application for the current paper. Section 4 examines previous works in the agricultural domain and compares them with the current agroString application. Section 5 gives the novel architecture for our current agroString application. The presentation of the algorithms that are used in agroString is given in Section 6. Sections 7 and 8 provide implementation details and the results for the CorDapp application used in the current paper. Finally, conclusions and suggestions for future improvements are discussed in Section 9.

## 2. Concerns and Challenges of Agricultural Production Distribution

A survey was conducted to elucidate the challenges and issues faced by agricultural produce distribution (Figure 3). It shows that 34% of the people surveyed were focused on improving provisions among the production facilities, 28 % were focused on improving traceability of the agricultural produce, and 20% were focused on decreasing the production time [12]. Here, we discuss some of the problems that can stem during agricultural production, processing, packaging, and distribution and due to the farmer's knowledge of technology [13].

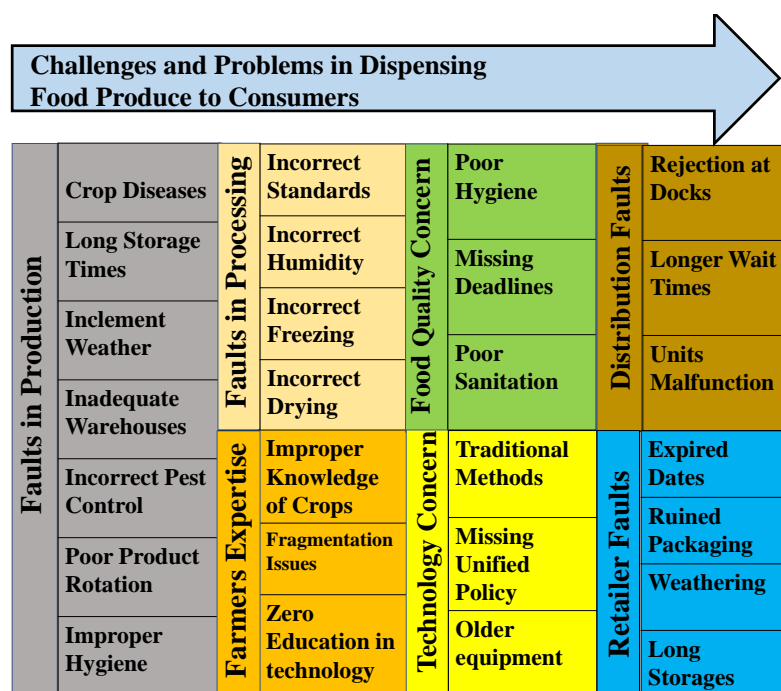


Source: Statista 2022

**Figure 3.** Statistics of consumer survey on challenges and issues faced by produce distribution.

Food quality is an essential concern in a supply chain because it is directly related to the well-being of individuals. The quantity of high-quality agricultural products available to the consumer can be unexpectedly reduced for various reasons at different stages of the supply chain, as shown in Figure 4. Circumstances for food quality degradation can occur due to unhealthy sanitation, unhygienic conditions, and missing deadlines between the supply chain participants. In food production, due to disease and inclement weather conditions, the crops can go unharvested and left in place to waste. Cosmetic imperfections of the produce can occur either during or after the harvest. Inefficient and long storage times after the harvest lead to production cost diminution. Storage issues can cause a lack of proper hygienic places for keeping the product nutritious and safe to consume. Due to inadequate warehousing facilities, wrong temperature, humidity, incorrect pest control procedures, and poor product rotations, the food quality can be degraded, making farmers face difficulties in getting reasonable prices from purchasers [14]. During the

processing of the products, freezing, drying, and slicing are standard measures for making any frozen foods or produce. If the procedure cannot maintain correct temperatures during manufacturing, it can lead to the spoilage of food.



**Figure 4.** Concerns and challenges of agricultural production distribution.

As farmers play a significant role in producing a healthy crop, their knowledge of technology usage on the field and active communication with food distributors will help in properly loading and transporting the produce toward end-users. However, inadequate knowledge and expertise in the current technologies have resulted in an increase in fragmenting issues from producers to intermediaries to consumers. With limited and inefficient technologies, the communication between the farmer and the intermediaries is complicated and uncertain. Using traditional methods and older equipment by the producers, it is hard to bring them under a single policy for the post-harvest phases in real-time. Longer waiting times, or rejection at the loading docks, malfunctioning of the refrigerating units, and accidents during truck transports can lead to an expiration of the produce shelf life. The long delays can result in ruined packaging and damages through weathering, which can lead to depreciated value and quality of the agricultural goods near the retail shops before they become available to consumers. A few practices, such as on-time delivery, maintaining hygienic requirements, and good sanitation standards, help increase the food's quality.

### 3. Novel Contributions

#### 3.1. Why Blockchain in Smart Agriculture?

The blockchain in smart farming plays an essential role in increasing trust in the data collected from inventories and farms. With the blockchain, communication and provenance of the goods can be provided to the consumers to verify product hygiene. This helps in avoiding food waste and keeps the product fresh until it reaches the end-users. With the blockchain introduced in agriculture, the information of the carriers, registering of the stakeholders and logistics involved, data regarding condition, price, and quality of the goods can be made visible, while also maintaining strict documentation for the products. With the blockchain, pricing imbalances can be avoided and the farmers have the full right to the price decisions on their supplies. Blockchain cryptocurrency can also be delivered as incentive to those participants/farmers who have optimally used resources and practiced

eco-friendly options for growing crops [15]. Figure 5 provides some of the use cases of the blockchain in smart agriculture.

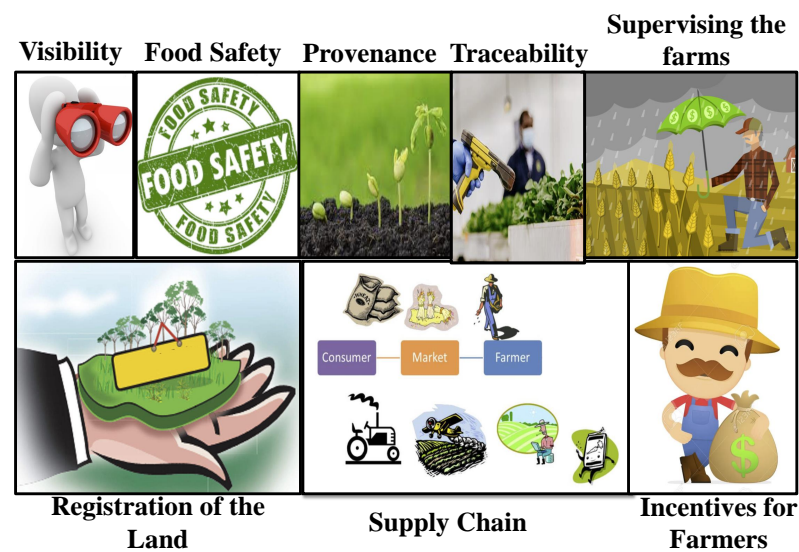


Figure 5. Blockchain use cases in smart agriculture.

Figure 6 shows the differences between private and public blockchains in the supply chain. With the use of a permissioned private enterprise blockchain distributed platform, the IoT limitations of data storage and security can be addressed to produce immutability, interoperability, and access control through smart contracts. Due to excessive resource consumption, traditional blockchain consensus mechanisms cannot be utilized with smart things. In the current paper, we have attempted to develop and implement a blockchain Corda-based data sharing framework for conventional agricultural produce distribution to the consumers by collecting the condition of the products at every stage with the help of the IoT.

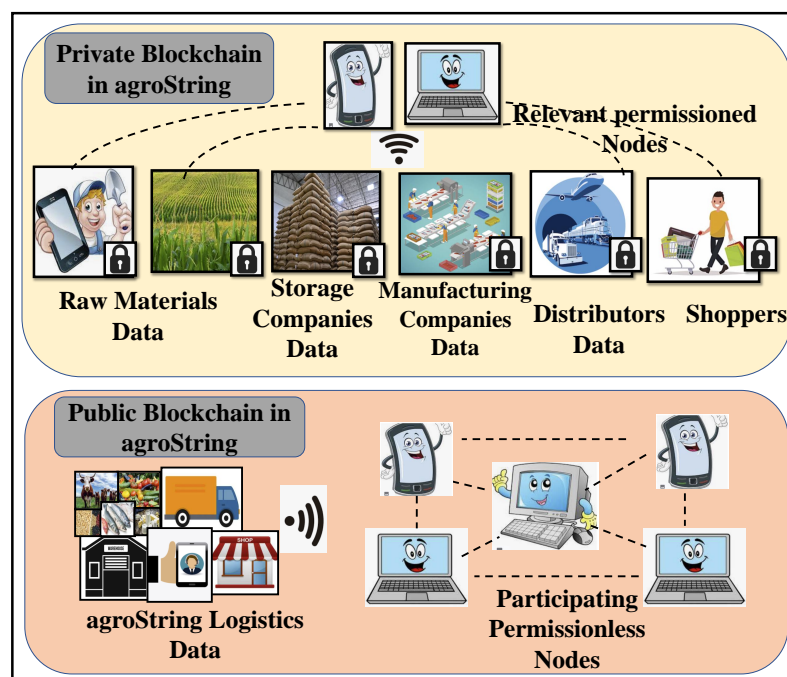


Figure 6. Comparison between private and public blockchains in supply chain.



### 3.2. Problems Addressed in the Current Paper

- Storage of data from the IoAT in central and cloud systems.
- Excessive transaction fees and mining time issues related to a public blockchain.
- Sharing of data to all the nodes that are participating.

### 3.3. Solutions Proposed in the Current Paper

- Evade centralized storage and implement decentralized storing and sharing.
- Use a private blockchain, also referred to as a permissioned blockchain.
- Propose a novel architecture for traceability and provenance in agroString.
- Reduced mining times.

### 3.4. Novelty and Significance of the Proposed Solutions

- Novel approach of distributed ledger technology for zero transaction fees (no cryptocurrency).
- Consistency and standards in communication between relevant parties with DeFi (Decentralized Finance) methodology for sharing the data transactions within permissioned peers with no intermediaries and within organization firewalls.
- A novel CorDapp private blockchain application that can be programmed.

## 4. Prior Related Work

In recent years, research and development work had been held between agricultural farmers and researchers. Various studies are being implemented to enhance mechanical and automatic methods for fusing information processing concepts and other fields.

With the help of Radio Frequency Identification (RFID), traceability is implemented for the fish supply chain in [16]. An architecture for live fish processing is designed and proposed for small enterprises. Each live fish has an RFID tag placed that connects it to logistic centers and retailers in order to provide the individual identities of the fish to the consumers. Sensors collect information during farming and transport. This work also makes use of web-based design for ease of use for both farmers and consumers.

To reduce the cost and load times of blockchain, a 2021 study [11] sent groundwater nitrate contamination data towards distributed storage in the Interplanetary File System (IPFS) and blockchain (Ethereum) for implementing dual hashing security and access control strategies.

In [17], an agro-food supply chain is proposed with the help of RFID for traceability with public blockchain technology. The distributed storage platform embedded in the system brings traceability with trusted information for the chain that enhances food safety throughout the supply process.

An IoT sensor was used for the tracking of cows in [18]. The paper presents a LoRaWAN architecture for communicating between long ranges and analyzes cattle tracking through a high-level system architecture. The protocols and application of the system are also further designed and developed.

For traceability in agriculture supply chain management, researchers [19] have executed a farm-to-fork blockchain. Both Ethereum (public blockchain) and Hyperledger Sawtooth (private blockchain) methods are implemented here. The performance is compared between both deployments for CPU and network usage, and their pros and cons are summarized.

Others [20] provide a theoretical representation of the whole supply chain network, starting from the provider to the consumer with the blockchain. The presented methodology attains traceability along with improved security, immutability, and faster transactions. The main objective is to provide transparency and accessibility to all the users of the supply network chain.

A traceability system [21] is designed with a dual storage structure, a blockchain as on-chain storage, and a traditional database for off-chain storage in order to decrease the load and cost and increase the ability to share traceability data safely.

The work in [22] uses the IoT to collect data and extends the application by embedding an Ethereum public blockchain to share data collected from the logistic stages securely. Through this, the system's performance is increased, securely transmitting the data and growing trust in data collected from relevant supply chain parties. Using drones in supply chain management is a recent IoAT progress in smart agriculture. Drones have many limitations when it comes to real-time usage, such as energy-draining, routing problems, and disposal effects [23]. The work in [24] introduces a system with blockchain data sharing to meet the issues of drones and proposes a mixed-integer programming model to formulate the intended problem and solve the problem with a tailored branch-and-price algorithm.

A plethora of research has been implemented for secured data flow for enterprise systems and health care using the blockchain, smart things, and similar techniques. From the IoT viewpoint, the blockchain has been implemented for enhancing data security and integrity through documenting health metrics in healthcare [25]. Contemplating the above literature, we have tried to develop a blockchain for the agri-dispense-data sharing system agroString between appropriate stakeholders. We have used IoAT-Edge device data and various records (Table 2, discussed in detail in Section 6) for averting manual recording, and we believe that using EBC technology and various sensors will improve real-time data sharing and augment security. The key point for Corda is that it can be employed to share basic text and numerical data between the relevant parties only. Table 1 shows a comparison of various agricultural data management applications with the current agro-String data management.

**Table 1.** Comparison of various agricultural data applications with agroString.

Application	Data Collection	Blockchain	Cost	Storage	Security
Fish Supplychain [16]	RFID	Not used	High	Centralized	Low
agro food Supplychain [17]	RFID	Ethereum	High	Decentralized	High
Cow Tracking [18]	IoT	Not Used	High	Centralized	Low
Agriculture Supplychain [19]	IoT	Ethereum and Hyperledger	Low	Decentralized	High
Agriculture Food Supplychain [20]-Theoretical	IoT	Ethereum	Low	Decentralized	High
Traceability System [21]	IoT	Ethereum	High	Centralized and Decentralized	High
Supplychain with Blockchain [22]	IoT	Ethereum	High	Decentralized	High
Blockchain with Drones for Supplychain [24]	Drones	Ethereum	High	Decentralized	High
agroString [Current-Paper]	IoT	Corda	Low	Decentralized	High

## 5. Architecture of the Proposed agroString

With the increasing demand and worldwide sourcing for food distribution for the global population, the safety and quality of agricultural produce has become a severe challenge. Some of the consequences of poor distribution can be related directly to the increase in product life-cycle cost, spoilage, and waste, along with bad efficiency. Here, efficiency is measured using the fraction of both transportation ( $T_n$ ) and distribution ( $D_n$ ). The capacity at which  $T_n$  and  $D_n$  can be achieved is calculated to obtain a snapshot of the food quality at a given time. The efficiency estimates for trucks and transports fall in between the range of 10 to 20%, and food spoilage and waste that are caused due to untimely delivery is approximately 12% [26]. More than 48 million people in the United states get sick due to food borne illness at least once every year [26].

To deal with and address some of the challenges of quality and timely delivery, the logistics need some technological facets that can provide tracking of the whole life cycle from providers and producers to costumers. The intermediaries include storage/warehousing, processing/packaging, transportation, and retailers/shops. The architecture shows the trail based on IoAT and EBC between the trusted parties. The flow of the data would be linear and only among those relevant.

### 5.1. Internet of Agriculture Things—Sensors and Networks for Quality Tracking and Communication

In the proposed architecture, the logistic stages that use the techniques for checking the temperature and humidity controls for agricultural goods are fixed and would require a more refined practical approach in real-time. By using the IoAT, the current condition of the produce and transmitting times of the product can be monitored. With the help of faster network connections, the data can be communicated in real-time and broadcast to the logistic strings [26].

For the warehouse and logistics phases, intelligent devices can be implanted to measure the quality. Every phase included in the food supply string can act as a communicating point, and the recorded flow of information would be provided to the retailers and consumers at the end for quality verification. We can achieve product tracking through sensing and communicating with the help of intelligent things. Continuously sensing temperature, humidity, and bacterial content accumulated on the produce [27] can reduce food deterioration with efficient timely remedial actions.

### 5.2. Private Blockchain—Achieving Access Control/Privacy/Trust in agroString

The blockchain is one of the fastest evolving technologies for secured data exchange and business operations. One of the leading platforms in this developing area is Corda. It takes the properties of a public blockchain, such as bitcoin and Ethereum, where anyone can initiate transactions, and fulfills the enterprise's requirements by inserting privacy and identity. Corda implements distributed, decentralized, permissioned, and open source Smart Contracts (SC). With the help of SC, access control, data integrity, privacy, and immutability are achieved [28].

Public and private blockchains are being used in supply chain applications to secure data transmitted between stakeholders. Blockchains have blocks connected in a chain pattern and utilize mining strategies for sending and receiving data. Corda, on the other hand, does not function with these approaches. In the present architecture, we demonstrate Corda-based private blockchain for receiving quality data of agricultural goods in the form of transactions near the communicating points. Corda transactions share the information between the connected logistic parties and charge zero transaction fees. Each transaction is provided with a tag containing secret codes belonging to the data that needs to be exchanged and shared between the relevant groups [28].

When digital files are reused, copied, and pasted several times, the value of the document rapidly becomes null, leading to double spending problems. Corda mitigates and prevents this issue by using a notary, deciding to sign if no problems arise and to not sign the transaction if double-spending occurs.

### 5.3. Consensus Mechanism—Corda Private Blockchain

Corda private blockchain does not use proof-of-work or proof-of-stake consensus mechanisms but still acts like a blockchain when adding a transaction to a ledger. The consensus is achieved by proving that the transaction is both valid and unique. Every transaction is a combination of states. Each state that is consumed is called input, and each state that is produced is referred to as output. When a state is consumed, it is like a money note being transferred to another party and will be marked as spent; each of these states carries a unique identifier, which is a combination of both Merkle hash and index number. For validity consensus, Corda checks that every transaction generated with input and output state is accepted by the smart contract of every input and output state and obtains all the required signatures for the transactions. This process is called walking the chain. The second is the uniqueness consensus, in which a notary checks for a node that has not used the same input state for multiple transactions. The transactions must achieve both validity and uniqueness consensus to be committed to the ledger in the Corda private blockchain.



#### 5.4. Architecture

Every node is unique and maintained by different institutes or companies belonging to agricultural stakeholders. In the architectural design presented in Figure 7, each logistic stage represents an independent node managed by their respective data centers connected via the Internet. The IoT devices are connected to each of these agricultural string logistics embedded with the Corda Shell application blockchain. The starting node contains an initiating flow that is given from the command-line CRaSH shell, which is a JVM-based instance. This flow helps start the entire echo process by sending the collected information from the IoAT to the recipient's node. The second node waits for the incoming data to send an acknowledgment by means of a responder flow. The shell acts as a bridge between the Corda container and its services within the JVM. After initiator and responder flows are written, they are packed, organized, and compiled in respective JAR files. These JAR files can be shared among the nodes that are germane. The communication between each logistic node is a similar process using Corda flows for tracking, saving, and forwarding immutable data towards consumers in the end.

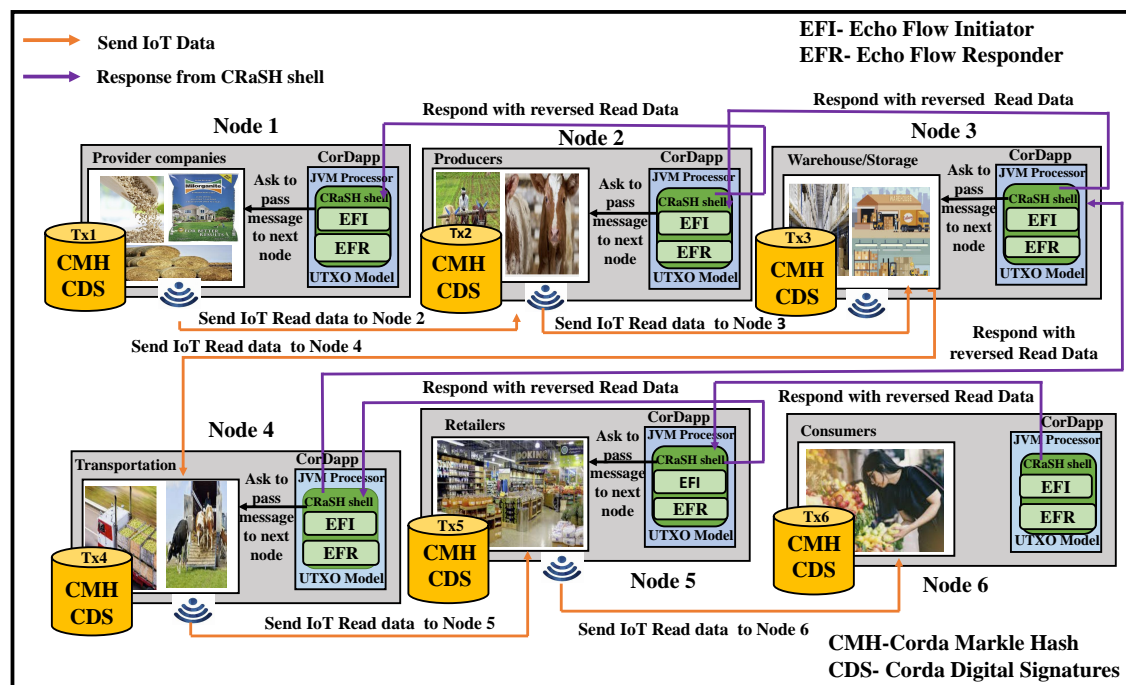


Figure 7. Proposed agroString architecture with IoAT and CorDapp.

#### 6. The Proposed Algorithms

Each logistic node represents the stakeholder present between the producer and consumer in the agricultural produce distribution. Each of these nodes sends requests for uploading IoAT (Datasets Table 2) data in real-time for signing and encrypting data. An elliptic curve mathematical function is used to generate private and public keys in these nodes. The uploaded data signifies a transaction that gets hashed and signed through the digital signatures exchanged among the nodes. Hash checks are performed to check for the integrity of the files. If the hashes are similar, the IoAT data gets signed successfully to start the flow of the data in between relevant nodes. The complete process for uploading and encrypting IoAT Data in CorDapp is given in Algorithm 1. Once the data gets uploaded to the node, the CorDapp encrypts using digital signatures distributed through the Certificate Authority (CA). The nodes participating exchange the certificates for sharing private and public keys to establish data legitimacy, valid proof of origin, and to make sure correct recipients obtain the data. As soon as the relevant node gets the encrypted data, the decryption process is initialized to check for the validity of the received data. Each node generates a public and private node in the decryption process, respectively. With the help of

the Merkle tree, the encrypted transactions present in the logistic nodes are retrieved. Each of these transaction signatures are compared to match the sender and receiver signatures for starting decryption. The CorDapp creates the flow if the decrypted transaction is similar to the uploaded transaction. A detailed flow of the decryption and retrieving of the file is given in Algorithm 2.

**Table 2.** Datasets for agroString.

Dataset Size	Data Name	Source	Link	Signed Transaction
701 KB	Supply chain logistics problem Data	Brunel University London.	Available online: <a href="https://brunel.figshare.com/articles/dataset/Supply_Chain_Logistics_Problem_Dataset/7558679/2">https://brunel.figshare.com/articles/dataset/Supply_Chain_Logistics_Problem_Dataset/7558679/2</a> (accessed on 1 August 2022)	7D5F62A5141BCCFCE851C7E1B9D974C0D0AD59B492DFD4FA20261485068694BB
516 KB	Livestock farming conditions Data	Kaggle	Available online: <a href="https://www.kaggle.com/datasets/jprukundo/ubudehelivestock1?resource=download">https://www.kaggle.com/datasets/jprukundo/ubudehelivestock1?resource=download</a> (accessed on 1 August 2022)	01CD8FBCAC33A0A88B7D6C1B4AF080F6EA8EDE32A902B2148C0694EA69571E87
12 KB	Fertilizer usage in Crops	USDA <sup>1</sup> & NASS <sup>2</sup>	Available online: <a href="https://www.nass.usda.gov/Surveys/Guide_to_NASS_Surveys/Chemical_Use/">https://www.nass.usda.gov/Surveys/Guide_to_NASS_Surveys/Chemical_Use/</a> (accessed on 1 August 2022)	2B19943EA812B0D1B9059E25D3F7F3D9CEEB94F76BC8CAE1E7620472A48DF0FE)
34 KB	Chemical usage in Diary	USDA & NASS	Available online: <a href="https://usda.library.cornell.edu/concern/publications/jh343s28d?locale=en">https://usda.library.cornell.edu/concern/publications/jh343s28d?locale=en</a> (accessed on 1 August 2022)	020431D918FCE620E0E66D315A808EE6552AFE23F662074F6F412047AFDF0375
177 KB	Cold Storage Data	USDA & NASS	Available online: <a href="https://usda.library.cornell.edu/concern/publications/pg15bd892?locale=en">https://usda.library.cornell.edu/concern/publications/pg15bd892?locale=en</a> (accessed on 1 August 2022)	CAB13B51E194029C303E9355BD25240E4D85B9BBAB26851AAE45560568CCA6D7
12.338 MB	Refrigerated Truck volumes data	USDA	Available online: <a href="https://agtransport.usda.gov/Truck/Refrigerated-Truck-Volumes/rfpn-7etz">https://agtransport.usda.gov/Truck/Refrigerated-Truck-Volumes/rfpn-7etz</a> (accessed on 1 August 2022)	DF34R4632R378645D703R766BD65789R8F23V7GGSW534781AA4578678TTA4DF
406 KB	Containerized grain Data	USDA & AMS <sup>3</sup>	Available online: <a href="https://agtransport.usda.gov/Container/Containerized-Grain-data/c353-2zjn">https://agtransport.usda.gov/Container/Containerized-Grain-data/c353-2zjn</a> (accessed on 1 August 2022)	0168135E8F56D02B6006114BBCD8E1E3A988077E6ACB0F42AF96D10E2D50F094
7.356 MB	Grain Inspection Data	USDA & AMS	Available online: <a href="https://agtransport.usda.gov/Exports/Grain-Inspections/sruw-w49i">https://agtransport.usda.gov/Exports/Grain-Inspections/sruw-w49i</a> (accessed on 1 August 2022)	392428FA9EDA1F8D40CC257F10FFD1AF83B4DBF089315F5880683DF6F4EAC1AE
15 KB	Temperature & Humidity Data	IoAT-Edge Device	IoAT-Edge Generated	4155092E577461253B2CE3FF1A9E990888536F51229576B27FD5C06FD529EB54

<sup>1</sup> U.S. Department of Agriculture. <sup>2</sup> National Agricultural Statistics Service. <sup>3</sup> Agricultural Marketing Service.

**Algorithm 1** Uploading and Encrypting IoAT Data in CorDapp.**Input:** IoAT Data(IoAT<sub>d</sub>)**Result:** Data signed and encrypted**Data:** Each Logistic node(L) generates Public and private Key pairs PuL, PrL respectively. $L_r \leftarrow IoAT_d$ 

/\* Request for IoAT Data \*/

 $PuL1 \leftarrow PrL1 \leftarrow Ec(F)$ 

/\* Private and Public keys are generated using elliptic curve(secp256r1) mathematical function \*/

 $Txn+ \leftarrow E[IoT_d signed] \leftarrow PuL_2 + PrL_1 + SHA_{256}(IoAT_d)$ 

/\* The Data gets hashed and signed with digital signatures exchanged between Logistic Nodes and added to the transaction \*/

/\* If hashes of IoATdata in first and second logistic nodes are similar, If IoAT data is signed successfully, start the data flow between relevant logistic nodes\*/

**if**  $L_2(SHA_{256}(IoAT_d)) == L_1(SHA_{256}(IoAT_d))$  **then**    **if** (IoAT<sub>d</sub>) is signed **then**         $L(Txn) \leftarrow CallEchoInitiatorFlow()$          $L(Txn) \leftarrow DeployEchoflows$          $L(Txn) \leftarrow Executeflows$ 

/\* Each Transaction once encrypted is added to Corda Markle tree for synchronizing and validating\*/

 $M_{tree}+ \leftarrow L(Txn)$     **end**

Discard Corda flow operations

**end**

End the process Repeat the steps for every IoT Data request.

**Algorithm 2** Accessing and Decrypting IoT Data.**Input:** Encrypted IoAT Data(E(IoAT<sub>d</sub>))**Result:** Data accessed and decrypted**Data:** Each Logistic node(L) generates Public and private Key pairs PuL, PrL respectively. $L_r \leftarrow E(IoAT_d)$ 

/\*A logistic node n requests for accesing Encrypted IoAT Data\*/

 $PuL1 \leftarrow PrL1 \leftarrow Ec(F)$ 

/\* Private and Public keys are generated using elliptic curve(secp256r1) mathematical function \*/

 $L(Txn) \leftarrow M_{tree}$ 

/\* The transaction in the logistic node is accessed from Markle tree \*/

/\* If the logistic node transaction signatures match with the sender and reciever signatures, Decrypt the transaction \*/

/\* If Dycrypted Transaction is similar to Uploaded Transaction data , Start Corda Flows \*/

**if**  $L(Txn)$  signatures match **then**     $D(L(Txn)) \leftarrow PrL_2 \leftarrow PuL_1 \leftarrow SHA_{256}(IoAT_d)$     **if**  $D(L(Txn)) == (L(Txn))$  **then**         $L(Txn) \leftarrow CallEchoResponderFlow()$          $L(Txn) \leftarrow Executeflows$     **end**

End the process

**end**

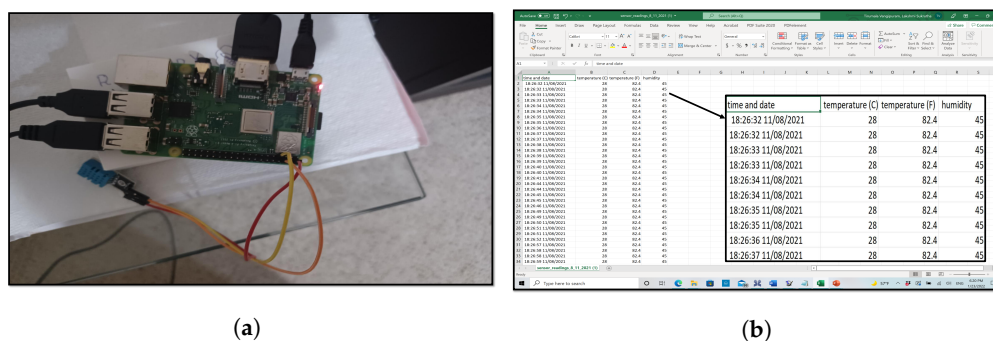
Decryption failed Repeat the steps for every encrypted IoT Data request.

## 7. Implementation of the Proposed Blockchain

For the current system, we have chosen Corda with java objects. The main point of using this system is that the data shared is only between relevant parties. Each node will send the transaction data along with tags of secret codes to make the sharing safer. With this secure hashing approach, each transaction avoids alterations to the data in later stages. The records are written in the format of Unspent Transaction Objects (UTXO) [28], which is the basic model of bitcoin. In order to represent states in transactions, we have written them as plain java objects. A notary's signature is embedded whenever there is a transaction. We have used JVM bytecode to define the structure and parameters of the state for more flexibility. Unlike other blockchain approaches, where transactions are broadcast to outsiders, Corda does not, because a notary prevents airing and double-spending and gives uniqueness to the transaction, as discussed in the mechanism of consensus in Corda in Section 5. The following implementation has been conducted in two phases.

### 7.1. Sensor Data

To perform phase one, we have collected temperature and humidity data using a sample IoAT-Edge device (DHT11 sensor + Raspberry Pi 3 Model B+) to obtain approximately 60,000 records, as shown in Figure 8. The gathered statistics were sent to the CorDapp application as a .zip file. Along with the real-time IoAT-Edge data, we also tested our agroString application with different datasets that are discussed in detail in Section 8.1.



**Figure 8.** Sensor data. (a) IoT-Edge device. (b) IoT-Edge device data in CSV format.

### 7.2. CorDapp agroString Application

In phase two, we cloned an already defined starter template to git for building our CorDapp. The call method in the `InitiatorFlow()` was used to write the application logic in java. With the help of `@InitiatedBy` annotation in the `ResponderFlow()`, we associated it with the `InitiatorFlow()`. The `InitiatorFlow` call method has different services for executing the logic. One of the identity services in this call method provides an address book from which we can retrieve the identity of the recipient and locate it. In our application, the message carried is the sensor data in the form of .csv converted to .zip. For matching the recipient's name, we used the `partiesFromName` method in the service to the recipient string. Once the recipient was matched, we received the list of party, and, using the `send()` method, we forwarded the sensor data to the following logistics recipient nodes.

To deploy and compile our CorDapp agroString application, we used Gradle. The `deployNodes` task in Gradle is mainly to name the nodes and configure them to a new folder. Each node has a name, port, and address. The naming convention of the Gradle uses X.500, where 'O' represents Organization, 'L' is a locality, and 'C' is for the country.

AgroString can take in any file format that can be uploaded and retrieved in a .zip attachment between any two nodes or parties. Node 1 sends an invoice to be received by Node 2 to download the attached .zip file to their local machine. A single state is used for the purpose of the invoice, and two flows are used for sending and downloading the files. The first flow sends and synchronizes the file between two participating logistic nodes. The files can be uploaded and attached from the local machine and can be retrieved through

the linked ID. The ID generated can be used for checking if the file downloaded is the same as the file uploaded and can help for signing the attachment.

## 8. Experimental Results

### 8.1. Datasets for *agroString*

The datasets are collected from different sources, as shown in Table 2. These datasets are used to test and compare the current CorDapp *agroString* application with existing applications.

#### 8.1.1. Supply Chain Logistics Problem Data

Many internal problems arise in the supply chain, such as production hazards, improper planning, incorrect data collection, technology issues, shipping limitations, communication problems, data integrity issues, and inefficient workflows. Management assessments are being conducted for identifying the risks present in the supply chain that impede the chain and also give scope to implement an immediate possible plan. The data set provides information regarding the problems related to carrying produce between ports using different transportation modes [29].

#### 8.1.2. Livestock Farming Conditions Data

It has been claimed that livestock are raised in confined and unsanitary conditions, with little personal space and breathing toxic gases, which results in respiratory issues in the animals. The sanitation of the confined space plays an essential role in livestock's health, hence the quality at the end. The livestock farming conditions database provides information on the geographic area where they are raised and the field's poverty status to differentiate the livestock's environmental setup [30].

#### 8.1.3. Fertilizer Usage in Crops

Fertilizers are mainly used to increase production and replace the nutrition that the soil loses by growing crops. Extensive usage of fertilizers can impact the environment (soil and water quality) and cause economic losses to the farmers. Excessive fertilization can impact soil and water nutrient balance and may affect crops. Fertilizer usage in the crops database gives information regarding how much fertilizer is being used for growing vegetables or other crops. With this data, the consumer can find if the amount and type of fertilizers used on the crops is within an acceptable range [31].

#### 8.1.4. Chemical Usage in Dairy

At dairy farms, the milk can have undesired ingredients due to grazing on the pesticide treated crops or because of insufficient disinfectants and detergents for cleaning operations. Most livestock farmers use veterinary drugs and other chemicals that may contain heavy metals, mycotoxins, and pesticides for efficiency in production, as well as for routine operation. These toxins and chemicals can be found in the milk, ultimately affecting the consumers [32].

#### 8.1.5. Cold Storage Data

Correct temperatures have to be maintained in every supply chain logistic stage for agricultural produce to avoid contamination, particularly in dairy and meat. Due to improper temperatures, environmental hazards and pathogen contamination can occur. The database for the cold storage has information regarding the regional and national monthly stocks of dairy, poultry, and meat products and the fruit and vegetables kept under private and semi-private refrigerated warehouses [33].

#### 8.1.6. Refrigerated Truck Volumes Data

For perishable goods (vegetables, frozen foods, ice cream, wine, etc.), different modes of transportation are being used, and trucks are one way of transferring these goods. However, for maintaining the proper hygiene of these perishable products, the transport



should be refrigerated for maintaining the correct temperatures. Refrigerated trucks are becoming more popular because of their economy, moving more goods in large volumes and enhancing ease of transport over long distances. The database for refrigerated truck volume is provided by the USDA/AMS/Market News/Specialty crops program movement, which includes the information on truck mode transport and imports at the domestic origins. The data provides daily fruit and vegetable refrigerated truck volumes since 2010, truck availability for transportation, and cost for hiring these trucks [34].

#### 8.1.7. Containerized Grain Data

Containerization is a process of carrying goods in containers that are of similar shape and size. Almost any type of goods can be stored in these containers and transported through rail, road, air, and ocean modes. Ocean shipping has become more popular for loading and unloading containers, with more frequent trips and minimum lost time near the ports. The containerized grain data here gives the movements of the United States waterborne grain exports between different origins and destinations [35].

#### 8.1.8. Grain Inspection Data

Grain inspection is a process that facilitates the marketing of agricultural products, such as meat, cereals, livestock, and fishery. It gives descriptions of the product for promoting honest trading and for benefiting the consumers. The grain inspection data contains enormous volumes of information regarding United States grains inspected for export from the U.S. port regions to the destination countries. The information saved here is the data related to exported grains examined under the authority of the U.S. Grain Standards Act [36].

### 8.2. Performance Testing-Private and Public Blockchain

JMeter Corda can be used for testing the performance of the Corda application flows. The Graphical User Interface (GUI) mode of the JMeter can be started as a client and used to create and view the test plans. The test plans can be generated by three components: Thread group, http request, and listener. We have tested agroSring dataset for private blockchain Corda, as shown in Figure 9, using JMeter and public blockchain, as given in Figure 10 through Ropsten Testnet.

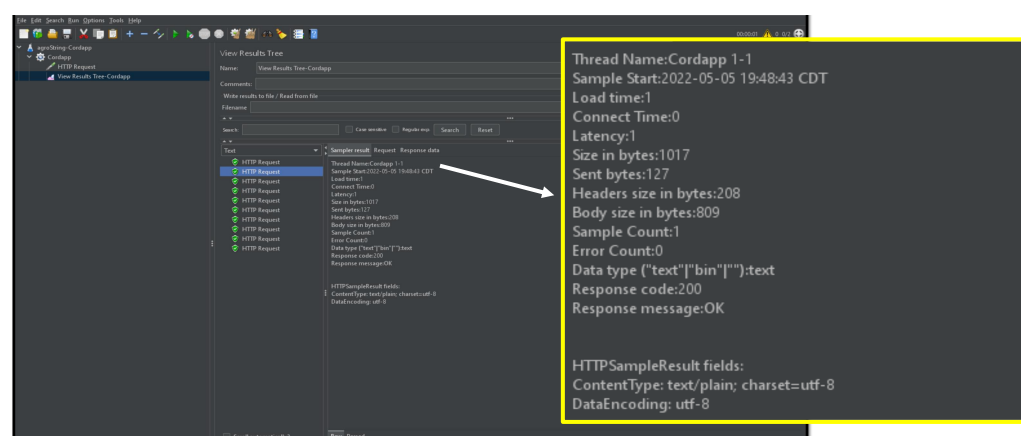


Figure 9. Performance Test for Corda Blockchain.

```

MINGW64~\Users\user\Documents\Blockchain\Coviddata
truffle migrate --network ropsten
Important!!
If you're using an HDWalletProvider, it must be web3 1.0 enabled or your migration will hang.

Starting migrations...
> Network name: 'ropsten'
> Network id: 3
> Block gas limit: 8000029

1. Initial_migration.js

Deploying 'Migrations'
> transaction hash: 0x8d660b734e200778603f7875710fde88e46e54cd9056bbe5ec378ec8f1c5d48
> Blocks: 0
> contract address: 0xa8f0d88ff62Aae230590a3f43AaaaF837c6A20D
> account: 0x7230637232f0951d01686f41cF43Cde1A683391
> balance: 0.92399468
> gas used: 230396
> gas price: 20 gwei
> value sent: 0 ETH
> total cost: 0.00460792 ETH

> Saving artifacts
> Total cost: 0.00460792 ETH

2. deploy_contracts.js

Replacing 'data'
> transaction hash: 0xfce18ef82bf26cb8a97415066b1457f2668139be00b25364f3e98814d93e440
> Blocks: 1
> contract address: 0x6d8ee41324f9b0Ec45aa45a234190E455Cd103FF
> account: 0x7230637232f0951d01686f41cF43Cde1A683391
> balance: 0.92399468
> gas used: 230396
> gas price: 20 gwei
> value sent: 0 ETH
> total cost: 0.00460792 ETH

> Saving artifacts
> Total cost: 0.00460792 ETH

Summary
> Total deployments: 2
> Final cost: 0.00950064 ETH

```

Figure 10. Performance Test for Public Blockchain.

Both record the load time, latency, and connecting times in milliseconds. The mining times and transaction costs are totally evaded in the Corda private blockchain, and Table 3 gives a comparison between prior works and the current agroString. The times and cost in the agro food supply chain [17] and agriculture supply chain [19] are calculated, assuming a traditional blockchain [37] to be 13.96 s [38] for 1MB Data and the cost to be \$1944.84 for one Ethereum [39], respectively.

Table 3. Comparison of prior works with current agroString.

Application	Blockchain	Latency	Off-Chain Storage	Transaction Cost	Financial Application
Fish Supplychain [16]	RFID	Not used	High	Centralized	Low
agro food	RFID	Ethereum	High	Decentralized	High
Supplychain [17]	IoT	Not Used	High	Centralized	Low
Cow Tracking [18]	Hyperledger	0.5 s	Used-Database	Hyperledger-No Cost	No
Traceability System [21]	Corda	1ms	Not Used	No Cost	Yes
agroString					
[Current-Paper]					
1 KB = 0.032 Eth [40] 1 MB = 32.768 1 Eth = 1944.84 [38]					

### 8.3. Why Corda Private Blockchain for agroString?

Different blockchain platforms, such as Ethereum, Hyperledger, and Corda are used for developing applications using distributed ledger technologies. Figure 11 shows a graph of transactions per second for each blockchain [41]. While all these blockchains benefit in terms of data integrity and security, they differ when it comes to vision and the field of application.

Both Ethereum and Hyperledger are valuable in various specific use cases, whereas Corda is most beneficial in applications related to the financial industry [42]. Producers make decisions in investment for farming mainly depending on the ease of convenience they get in accessing financial means. When the financial instruments cannot fulfill the farmer's requirements and provide reduced-risk products, farmers are not interested in using new technologies and updated methods to increase their financial stability [43,44]. Figure 12 shows how agricultural finance influences yield in the farm sector. There are two types of financing in agriculture: *Value chain financing* is the number of steps handled by the actors or stakeholders involved in bringing the food produce to the end consumers by adding value at each stage of the product without including banks. In this financing, the actors involved provide loans and add value to the products in each logistic step. With this financing, the farmer is at an advantage in getting new types of equipment and

technologies for a good yield, resulting in a reasonable price for the product and increased farmer's economic status. *Direct* financing involves banks that give financial loans to the farmers for a certain period of repaying after the harvest.

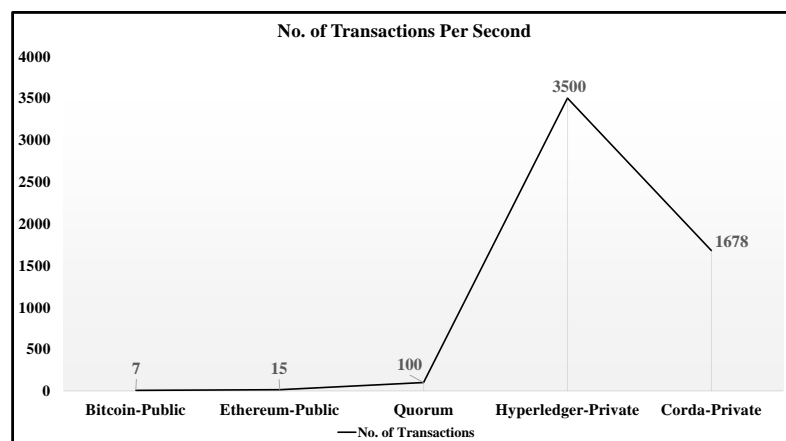


Figure 11. Number of transactions per second in different blockchains.

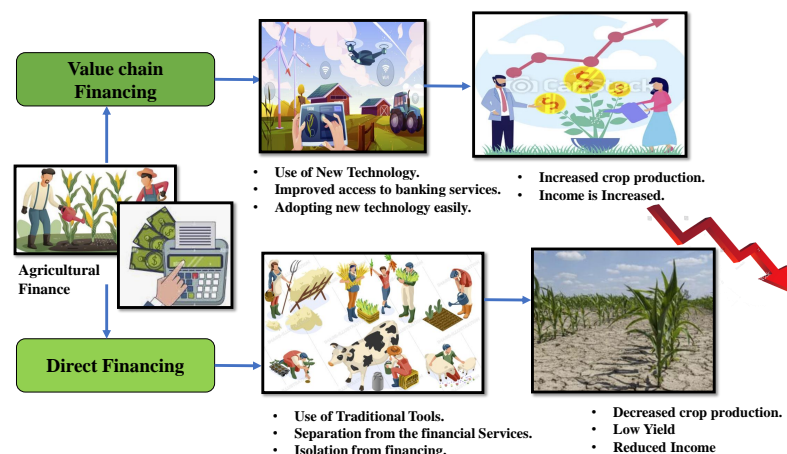


Figure 12. Impact of Agricultural Finance on Agricultural yield.

A simple supply chain, with direct financing, moves data collected from the supplier to the consumer, but a supply chain embedded with a value chain financing concept adds value along the chain, both to the product and the stakeholders involved. Therefore, we take advantage of the Corda framework to combine the agriculture finance feature into our current agroString application.

#### 8.4. Results

We deployed and ran the nodes as depicted in Figure 13a,b. Once the nodes started to run, the flow was initiated from source Node 1 to Node 2 receiver. The transaction to attach the .zip file was created and processed. Upon completing the process, the notary was used to sign and record the transaction and also obtain the counterparties' signatures. After collecting all the signatures, they were verified, and the transaction was broadcast to the participant logistic nodes. The flow ended once the total transaction was given an ID. The entire flow is shown in Figure 13c.

The ID given from the flow start was used to check the file's authenticity and correct origin. Here, the .zip file was attached to the receiver Node in the flow start. To retrieve the .zip file, node 2 checks for the attachment ID and verifies it to download the zip file, as shown in Figure 13d. Thus, the sensor readings and datasets were successfully transferred between relevant parties while maintaining data integrity and quality with zero transaction fees and validity times in the corDapp environment. The application takes in

any file format and up to 10 MB size of the file in the form of a .zip attachment. The entire application phases are shown in Figure 13.

```
sukrutha@DESKTOP-DC67C70 MINGW64 ~/OneDrive/Documents/samples/sendfile-Attachments (release-V4)
$ ./gradlew Attachments:gradlew workflows-java:deployNodes
bash: sendfile-Attachments/gradlew: No such file or directory

sukrutha@DESKTOP-DC67C70 MINGW64 ~/OneDrive/Documents/samples/sendfile-Attachments (release-V4)
$ ./gradlew workflows-java:deployNodes
Starting a Gradle Daemon (subsequent builds will be faster)

*[2A+][1B+][1m] Starting Daemon-[m+][7D+][1B+][1A+] IDLE-[0K+][6D+][1B+][2A+][1m<+][0;1m-----> 0% INITIALIZING [0s]+[m+][36D+][1B+][1m] Eval
uating settings-[m+][2D+][1B+][2A+][1m<+][0;1m-----> 0% CONFIGURING [0s]+[m+][0K+][35D+][1B+][1m] Loading projects+[m+][0K+][18D+][1B+][1A+
][1m] root project-[m+][0K+][14D+][1B+][2A+][1m<+][0;1m-----> 0% CONFIGURING [1s]+[m+][35D+][2B+][1A+][1m] root project > Resolve dependen
cies of :classpath+[m+][51D+][1B+][1A+][1m] root project > Resolve files of :classpath+[m+][0K+][44D+][1B+][1A+][1m] root project-[m+][0K+][14D+][1
B+][2A+][1m<+][0;1m-----> 0% CONFIGURING [2s]+[m+][35D+][2B+][1A+][1m<+][0;32;1m=====+][0;39;1m-----> 20% CONFIGURING [2s]+[m+][36D+][1
B+][1m] :contracts-java+[m+][17D+][1B+][2A+][1m<+][0;32;1m=====+][0;39;1m-----> 60% CONFIGURING [2s]+[m+][36D+][1B+][1m] :workflows-java+[m+][1
7D+][1B+][2A+][1m<+][0;32;1m=====+][0;39;1m] 100% CONFIGURING [2s]+[m+][37D+][1B+][1DLE-[0K+][6D+][1B+][2A+][1m<+][0;1m-----> 0% EXE
CUTING [2s]+[m+][0K+][33D+][1B+][1m] :contracts-java:compileKotlin > Resolve dependencies of :contracts-java:compileClasspath+[m+][90D+][1B+][1
A+][1m] :contracts-java:compileKotlin+[m+][0K+][31D+][1B+][2A+][1m<+][0;1m-----> 6% EXECUTING [2s]+[m+][33D+][1B+][1m] :contracts-java:c
ompileJava+[m+][0K+][29D+][1B+][2A+][1m<+][0;32;1m=====+][0;39;1m-----> 46% EXECUTING [3s]+[m+][34D+][1B+][1m] :workflows-java:compileKotlin >
Resolve dependencies of :workflows-java:compileClasspath+[m+][90D+][1B+][2A+][1m<+][0;32;1m=====+][0;39;1m-----> 73% EXECUTING [3s]+[m+][3
0D+][1B+][1m] :workflows-java:configureCordappFaker > Resolve dependencies of :workflows-java:runtime+[m+][90D+][1B+][2A+][1m<+][0;32;1m=====+][0;39;1m-> 93% EXECUTING [3s]+[m+][34D+][1B+][1m] :workflows-java:deployNodes+[m+][0K+][29D+][1B+][2A+][1m<+][0;32;1m=====+][0;39;
1m-> 93% EXECUTING [4s]+[m+][34D+][2B+][2A+][1m<+][0;32;1m=====+][0;39;1m-> 93% EXECUTING [5s]+[m+][34D+][2B+][2A+][0K
+][1m] Task :workflows-java:deployNodes+[m
Running Condforn task
Deleting C:\Users\sukrutha\OneDrive\Documents\samples\sendfile-Attachments\workflows-java\build\nodes
Bootstrapping local test network in C:\Users\sukrutha\OneDrive\Documents\samples\sendfile-Attachments\workflows-java\build\nodes
Generating node directory for Buyer
Generating node directory for Notary
Generating node directory for Seller
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Waiting for all nodes to generate their node-info files...
*[0K
*[0K
*[2A+][1m<+][0;32;1m=====+][0;39;1m-> 93% EXECUTING [5s]+[m+][34D+][1B+][1m] :workflows-java:deployNodes+[m+][29D+][1B
```

(a)

```
[32]:imBUILD SUCCESSFUL [0:39m in 29s]
12 actionable tasks: 3 executed, 9 up-to-date
+@OK
+@OK
*[2A+[1mc:[0;1m----- 0% WAITING[m=[26D[1B-[1m :workflows-java:deployNodes+[m=[29D+[1B-[2A+[2K+[1B-[2K+[1A
sukrutha@DESKTOP-DC67C70 MINGW64 ~/OneDrive/Documents/samples/sendfile-Attachments (release-V4)
$ . /workflows-java/build/nodes/runnodes
Starting nodes in C:\Users\sukrutha\OneDrive\Documents\samples/sendfile-Attachments\workflows-java\build\nodes
No file corda.jar found in C:\Users\sukrutha\OneDrive\Documents\samples/sendfile-Attachments\workflows-java\build\nodes
Starting corda.jar in C:\Users\sukrutha\OneDrive\Documents\samples/sendfile-Attachments\workflows-java\build\nodes
5005
Node will expose jolokia monitoring p
Running command: cmd /C start C:\Program Files\Java\jdk8.0.30\tomcat\bin\java.exe -Xmx512M -jar "C:\Users\sukrutha\OneDrive\Documents\samples/sendfile-Attachments\workflows-java\build\nodes\corda.jar" --port=5005
Listening for transport dt_socket at address: 5006
 When I discovered my toaster wasn't waterproof, I was shocked.
----- Corda Open Source 4.4 (21e8c4f) -----
Logs can be found in %USERPROFILE%\AppData\Local\Corda\logs
P2pJolokia: Agent started with URL http://127.0.0.1:7006/jolokia/
Warning! This node is running in development mode! It is not safe for production deployment.
Advertised P2P messaging addresses: *: localhost:10002
Server=y,suspendn,address=5006" "-jaVnc connection address *: localhost:10003
after -Name=Notary -jar C:\Users\sukrutha\OneDrive\Documents\samples/sendfile-Attachments\workflows-java\build\nodes\corda.jar --port=5006 Loaded Z corbbpp(s) *: localhost:10004
Starting corda.jar in C:\Users\sukrutha\OneDrive\Documents\samples/sendfile-Attachments\workflows-java\build\nodes
5007 licence Apache License, Version 2.0, workflow CorDbapp: sendfile Flows version 1 by vendor Corda Open Source with licenc
e Apache License, Version 2.0
Node for "Notary" started up and registered in 20:92 sec
Planning P2PMessaging loop
Node will expose jolokia monitoring
Running command: cmd /C start C:\Program Files\Java\jdk8.0.30\tomcat\bin\java.exe -Xmx512M -jar "C:\Users\sukrutha\OneDrive\Documents\samples/sendfile-Attachments\workflows-java\build\nodes\corda.jar" --port=5007
Server=y,suspendn,address=5007" "-jaWelcome to the Corda interactive shell. You can see the available commands by typing 'help'.
after -Name=Seller -jar C:\Users\sukrutha\OneDrive\Documents\samples/sendfile-Attachments\workflows-java\build\nodes\corda.jar --port=5008
Started 3 processes
Finished starting nodes
Tue Jan 11 11:43:58 CST 2022>>>
Node for buyer started up and registered in 20:90 sec
Running P2PMessaging loop
Welcome to the Corda interactive shell. You can see the available commands by typing 'help'.
Tue Jan 11 11:43:58 CST 2022>>>
You can see the available commands by typing 'help'.
Tue Jan 11 11:43:59 CST 2022>>>
```

(b)

Figure 13. *Cont.*

```

Listening for transport dt_socket at address: 5007

Corda
Moses had the first tablet that
could connect to the cloud.

--- Corda Open Source 4.4 (21e8c4f) ---

Logs can be found in : C:\Users\sukrutha\OneDrive\Documents\samples\sendfile-Attachments\workflows-java\build\nodes\Seller\logs
Jolokia: Agent started with URL http://127.0.0.1:7007/jolokia/
! ATTENTION: This node is running in development mode! This is not safe for production deployment.
Advertised P2P messaging addresses : localhost:10008
RPC connection address : localhost:10009
RPC admin connection address : localhost:10049
Loaded 2 CorDapp(s) : Contract CorDapp: sendfile Contract version 1 by vendor Corda Open Source with licence Apache License, Version 2.0, Workflow CorDapp: sendfile Flows version 1 by vendor Corda Open Source with licence Apache License, Version 2.0
Node for "Seller" started up and registered in 18.44 sec
Running P2P messaging loop

Welcome to the Corda interactive shell.
You can see the available commands by typing 'help'.

Tue Jan 11 11:43:59 CST 2022>>> flow start SendAttachment receiver: Buyer
Working Directory = C:\Users\sukrutha\OneDrive\Documents\samples\sendfile-Attachments\workflows-java\build\nodes\Seller
Starting
Generating transaction
PROCESS transaction
Obtaining notary signature and recording transaction.
Collecting signatures from counterparties.
Verifying collected signatures.
Obtaining notary signature and recording transaction.
Broadcasting transaction to participants
Done
Flow completed with result: SignedTransaction(id=C96233B5F52763DF8A8CDF501C9A233278FE1CA55F142BFE7D8C446CF2F86A)

Tue Jan 11 11:46:29 CST 2022>>>

```

(c)

```

Listening for transport dt_socket at address: 5005

Corda
You should really try a seafood diet.
It's easy: you see food and eat it.

--- Corda Open Source 4.4 (21e8c4f) ---

Logs can be found in : C:\Users\sukrutha\OneDrive\Documents\samples\sendfile-Attachments\workflows-java\build\nodes\Buyer\logs
! ATTENTION: This node is running in development mode! This is not safe for production deployment.
Jolokia: Agent started with URL http://127.0.0.1:7005/jolokia/
Advertised P2P messaging addresses : localhost:10005
RPC connection address : localhost:10006
RPC admin connection address : localhost:10046
Loaded 2 CorDapp(s) : Contract CorDapp: sendfile Contract version 1 by vendor Corda Open Source with licence Apache License, Version 2.0, Workflow CorDapp: sendfile Flows version 1 by vendor Corda Open Source with licence Apache License, Version 2.0
Node for "Buyer" started up and registered in 17.04 sec
Running P2P messaging loop

Welcome to the Corda interactive shell.
You can see the available commands by typing 'help'.

Tue Jan 11 11:43:58 CST 2022>>> flow start DownloadAttachment sender: Seller, path: file.zip
Starting
Retrieving attachment ID
Download attachment
Done
Flow completed with result: Downloaded file from Seller to file.zip

Tue Jan 11 11:52:57 CST 2022>>> flow start DownloadAttachment sender: Seller, path: sensor_readings.zip
Starting
Retrieving attachment ID
Download attachment
Done
Flow completed with result: Downloaded file from Seller to sensor_readings.zip

Tue Jan 11 11:53:43 CST 2022>>>

```

(d)

**Figure 13.** agroString CorDapp Application. (a) Deploying nodes with Gradle. (b) Running nodes. (c) Flows start for attachment of file. (d) Retrieving the file.

## 9. Conclusions and Future Direction

The agroString application is successfully built in two phases, one to generate the data from the raspberry pi and the second for storing and retrieving the generated files and datasets given in Table 2. This paper illustrates a novel architecture for the supply chain using Corda Shell and uses datasets for reading in real-time. The CorDapp agroString design is mainly based on a financial flow application that addresses the central and different blockchain system limitations. A comparative analysis of the existing approaches to the current agroString has been illustrated to show that our method gives higher data integrity and evades transactional cost and latency issues, providing more privacy and security to the data. In the future, we believe that combining both phases of IoAT into a CorDapp application can improve the agroString performance, further decreasing IoAT and blockchain challenges and enhancing security levels.

**Author Contributions:** Methodology, S.L.T.V.; Project administration, S.P.M.; Writing—review & editing, E.K. and C.R. All authors have read and agreed to the published version of the manuscript.



**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Conflicts of Interest:** The authors declare that they have no conflict of interest.

## References

1. UDSA. With Adequate Productivity Growth, Global Agriculture Is Resilient to Future Population and Economic Growth. 2014. Available online: <https://www.ers.usda.gov/amber-waves/2014/december/with-adequate-productivity-growth-global-agriculture-is-resilient-to-future-population-and-economic-growth/> (accessed on 13 March 2022).
2. NRDC. Wasted: How America Is Losing up to 40 Percent of Its Food from Farm to Fork to Landfill. 2017. Available online: <https://www.nrdc.org/sites/default/files/wasted-2017-report.pdf> (accessed on 9 January 2022).
3. Hafeez, A. Role of Quality in Supply Chain Management. 2014. Available online: <https://supply-chain.cioreview.com/cxinsight/role-of-quality-in-supply-chain-management-nid-4519-cid-78.html> (accessed on 19 March 2022).
4. Mitra, A.; Vangipuram, S.L.T.; Bapatla, A.K.; Bathalapalli, V.K.V.V.; Mohanty, S.P.; Kougianos, E.; Ray, C. Everything You wanted to Know about Smart Agriculture. *arXiv* **2022**, arXiv:2201.04754.
5. Popova, K. IoT Challenges Associated With the Agriculture Industry. 2018. Available online: <https://www.techtarget.com/iotagenda/blog/IoT-Agenda/IoT-challenges-associated-with-the-agriculture-industry> (accessed on 19 March 2022).
6. Pallagani, V.; Khandelwal, V.; Chandra, B.; Udutalapally, V.; Das, D.; Mohanty, S.P. dCrop: A Deep-Learning Based Framework for Accurate Prediction of Diseases of Crops in Smart Agriculture. In Proceedings of the 2019 IEEE International Symposium on Smart Electronic Systems, Rourkela, India, 16–18 December 2019; pp. 29–33. [CrossRef]
7. Kumar, S.; Chowdhary, G.; Udutalapally, V.; Das, D.; Mohanty, S.P. gCrop: Internet-of-Leaf-Things (IoLT) for Monitoring of the Growth of Crops in Smart Agriculture. In Proceedings of the 2019 IEEE International Symposium on Smart Electronic Systems, Rourkela, India, 16–18 December 2019; pp. 53–56. [CrossRef]
8. Sethuraman, S.; Tadkapally, G.; Mohanty, S.; Subramanian, A. iDrone: IoT-Enabled Unmanned Aerial Vehicles for Detecting Wildfires Using Convolutional Neural Networks. *SN Comput. Sci.* **2022**, *3*, 1–13. [CrossRef]
9. Koopairojn, S.; Puitrakul, C.; Bangkok, T.; Riyagoon, N.; Ruengittinun, S. Smart tag tracking for livestock farming. In Proceedings of the 2017 10th International Conference on Ubi-media Computing and Workshops, Pattaya, Thailand, 1–4 August 2017; pp. 1–4. [CrossRef]
10. Bapatla, A.K.; Mohanty, S.P.; Kougianos, E. sFarm: A Distributed Ledger Based Remote Crop Monitoring System for Smart Farming. In *IFIP International Internet of Things Conference*; Springer: Cham, Switzerland, 2021; pp. 13–31. [CrossRef]
11. Sukrutha L. T., V.; Mohanty, S.P.; Kougianos, E.; Ray, C. G-DaM: A Blockchain based Distributed Robust Framework for Ground Water Data Management. In Proceedings of the 2021 IEEE International Symposium on Smart Electronic Systems, Jaipur, India, 18–22 December 2021; pp. 261–266. [CrossRef]
12. Mazareanu, E. Top Strategic Visibility Focus in Production in Global Supply Chains for 2017. 2018. Available online: <https://www.statista.com/statistics/856577/strategic-visibility-focuses-production-supply-chain/> (accessed on 14 March 2022).
13. Negi, S.; Anand, N. Issues and Challenges in the Supply Chain of Fruits & Vegetables Sector in India: A Review. *Int. J. Manag. Value Supply Chain. (IJMVSC)* **2015**, *6*, 47–62. [CrossRef]
14. McCartney, I. 6 Common Problems with Warehouse Food Storage. 2018. Available online: <https://kempner.co.uk/2018/09/06/6-common-problems-with-warehouse-food-storage/> (accessed on 14 March 2022).
15. Writer, E.G. Blockchain in Agriculture: How Crypto Is Disrupting Farming. 2021. Available online: <https://stories.pinduoduo-global.com/agritech-hub/blockchain-in-agriculture> (accessed on 16 March 2022).
16. Hsu, Y.C.; Chen, A.P.; Wang, C.H. A RFID-enabled traceability system for the supply chain of live fish. In Proceedings of the 2008 IEEE International Conference on Automation and Logistics, Qingdao, China, 1–3 September 2008; pp. 81–86. [CrossRef]
17. Tian, F. An agri-food supply chain traceability system for China based on RFID & blockchain technology. In Proceedings of the 2016 13th International Conference on Service Systems and Service Management, Kunming, China, 24–26 June 2016; pp. 1–6. [CrossRef]
18. Zinas, N.; Kontogiannis, S.; Kokkonis, G.; Valsamidis, S.; Kazanidis, I. Proposed open source architecture for Long Range monitoring. The case study of cattle tracking at Pogoniani. In Proceedings of the 21st Pan-Hellenic Conference on Informatics, Larissa, Greece, 28–30 September 2017; pp. 1–6. [CrossRef]
19. Caro, M.P.; Ali, M.S.; Vecchio, M.; Gialfreda, R. Blockchain-based traceability in Agri-Food supply chain management: A practical implementation. In Proceedings of the 2018 IoT Vertical and Topical Summit on Agriculture, Tuscany, Italy, 8–9 May 2018; pp. 1–4. [CrossRef]
20. Madumidha, S.; Ranjani, P.S.; Vandhana, U.; Venmuhilan, B. A Theoretical Implementation: Agriculture-Food Supply Chain Management using Blockchain Technology. In Proceedings of the 2019 TEQIP III Sponsored International Conference on Microwave Integrated Circuits, Photonics and Wireless Networks, Tiruchirappalli, India, 22–24 May 2019; pp. 174–178. [CrossRef]
21. Yang, X.; Li, M.; Yu, H.; Wang, M.; Xu, D.; Sun, C. A Trusted Blockchain-Based Traceability System for Fruit and Vegetable Agricultural Products. *IEEE Access* **2021**, *9*, 36282–36293. [CrossRef]

22. Pradhan, N.R.; Singh, A.P.; Mahule, R. Blockchain Based Smart and Secure Agricultural Monitoring System. In Proceedings of the 2021 5th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 22–23 October 2021; pp. 1–6. [CrossRef]
23. McCloud, D.C. The Future of Drones in Logistics and Supply Chain Management. 2022. Available online: <https://www.shippingsolutions.com/blog/the-future-of-drones-in-logistics-and-supply-chain-management> (accessed on 20 October 2022).
24. Xia, Y.; Zeng, W.; Xing, X.; Zhan, Y.; Tan, K.H.; Kumar, A. Joint optimisation of drone routing and battery wear for sustainable supply chain development: A mixed-integer programming model based on blockchain-enabled fleet sharing. *Ann. Oper. Res.* **2021**. [CrossRef]
25. Rachakonda, L.; Mohanty, S.P.; Kougianos, E. Stress-Lysis: An IoMT-Enabled Device for Automatic Stress Level Detection from Physical Activities. In Proceedings of the 2020 IEEE International Symposium on Smart Electronic Systems, Chennai, India, 14–16 December 2020; pp. 204–205. [CrossRef]
26. Pal, A.; Kant, K. IoT-Based Sensing and Communications Infrastructure for the Fresh Food Supply Chain. *Computer* **2018**, *51*, 76–80. [CrossRef]
27. Donlon, M. Sensor Detects Pesticides, Bacteria on Fruits and Vegetables. 2021. Available online: <https://insights.globalspec.com/article/17476/sensor-detects-pesticides-bacteria-on-fruits-and-vegetables> (accessed on 20 March 2022).
28. Sheikh, J. *Mastering Corda*; Oreilly and Associates Inc.: Sebastopol, CA, USA, 2020.
29. Kalganova, T.; Dzalbs, I. *Supply Chain Logistics Problem Dataset*; Brunel University: London, UK, 2019. [CrossRef]
30. Rukundo, J.P. Ubudehe Livestock 1. 2019. Available online: <https://www.kaggle.com/datasets/jprukundo/ubudehelivestock1?resource=download> (accessed on 11 December 2021).
31. USDA. Agricultural Chemical Use Program. 2022. Available online: [https://www.nass.usda.gov/Surveys/Guide\\_to\\_NASS\\_Surveys/Chemical\\_Use](https://www.nass.usda.gov/Surveys/Guide_to_NASS_Surveys/Chemical_Use) (accessed on 8 January 2022).
32. USDA and NASS. Economics, Statistics and Market Information System. 2022. Available online: <https://usda.library.cornell.edu/concern/publications/jh343s28d?locale=en> (accessed on 10 February 2022).
33. USDA and NASS. Cold Storage. 2022. Available online: <https://usda.library.cornell.edu/concern/publications/pg15bd892?locale=en> (accessed on 15 December 2021).
34. USDA. Refrigerated Truck Volumes. 2022. Available online: <https://agtransport.usda.gov/Truck/Refrigerated-Truck-Volumes/rfpn-7et> (accessed on 15 December 2021).
35. USDA and AMS. Containerized Grain Data . 2022. Available online: <https://agtransport.usda.gov/stories/s/U-S-Waterborne-Containerized-Grain-Exports/33uv-zvv7/> (accessed on 28 January 2022).
36. USDA and AMS. Grain Inspections. 2022. Available online: <https://agtransport.usda.gov/Exports/Grain-Inspections/sruw-w49i> (accessed on 8 January 2022).
37. Frankenfield, J. Block Time. 2021. Available online: <https://www.investopedia.com/terms/b/block-time-cryptocurrency.asp> (accessed on 13 March 2022).
38. YCHARTS. Ethereum Average Block Time. 2022. Available online: [https://ycharts.com/indicators/ethereum\\_average\\_block\\_time](https://ycharts.com/indicators/ethereum_average_block_time) (accessed on 13 March 2022).
39. YCharts. Ethereum Price. 2022. Available online: [https://ycharts.com/indicators/ethereum\\_price](https://ycharts.com/indicators/ethereum_price) (accessed on 27 March 2022).
40. Wood, D.G. Ethereum: A Secure Decentralised Generalised Transaction Ledger. 2014. Available online: <https://gavwood.com/paper.pdf> (accessed on 27 March 2022).
41. Corda. Transactions Per Second (TPS). 2018. Available online: <https://www.corda.net/blog/transactions-per-second-tps/> (accessed on 11 January 2022).
42. Geroni, D. Hyperledger vs. Corda vs. Ethereum: The Ultimate Comparison. 2021. Available online: <https://101blockchains.com/hyperledger-vs-corda-r3-vs-ethereum/#prettyPhoto> (accessed on 11 January 2022).
43. Ruiz, C. How Can Finance Influence Productivity of Agricultural Firms? 2014. Available online: <https://blogs.worldbank.org/allaboutfinance/how-can-finance-influence-productivity-agricultural-firms> (accessed on 20 October 2022).
44. Bank, T.W. Agriculture Finance & Agriculture Insurance. 2020. Available online: <https://www.worldbank.org/en/topic/financialsector/brief/agriculture-finance> (accessed on 27 March 2022).