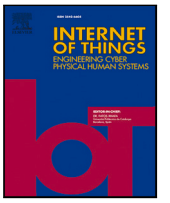
Internet of Things 26 (2024) 101215

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot



Research article

Secure pharmaceutical supply chain using blockchain in IoT cloud systems

Mangala N. [a,*], Naveen D.R. [b], B. Eswara Reddy [a], Rajkumar Buyya [c,1], Venugopal K.R. [b,1], S.S. Iyengar [d,1], L.M. Patnaik [e,1]

[a] Computer Science and Engineering, JNTU Anantapur, Anantapuramu, 515002, Andra Pradesh, India
[b] Computer Science and Engineering, UVCE, Bangalore University, Bangalore, 560001, Karnataka, India
[c] Cloud Computing and Distributed Systems Laboratory, University of Melbourne, Melbourne, 3052, Parkville, Australia
[d] Computer Science, Florida International University, Miami, 32306, FL, USA
[e] Consciousness Studies Program, National Institute of Advanced Studies, Bangalore, 560054, Karnataka, India

ARTICLE INFO

Keywords:
Internet of Things (IoT)
Blockchain
Supply Chain Management (SCM)
Cloud computing (CC)
Artificial Intelligence (AI)

ABSTRACT

Supply Chain Management (SCM) systems require time sequencing, coordination and tracking the movement of goods and processes. Internet of Things (IoT) and Blockchain technologies are useful to develop a secure automated SCM. IoT devices with in-built sensors and actuators help to keep track of the state, location, temperature or other parameters of an entity, and control the automation of routine as well as hazardous tasks. Blockchain technology supports time-stamping, authentication, process coordination, non-repudiation, commercial transactions, and also provides security for transactions and storage. The pharmaceutical SCM demands accurate, immediate and secure control system. Additionally, the supply chain process data from IoTs is stored and processed in Cloud by Analytics applications, for business planning and improvement. An efficient and secure IoT-Cloud-Blockchain based system for both SCM automation and analytics has been proposed in this work. It leverages a hierarchical IoT, Mist, Edge, Fog, Cloud computing (IMEFC) architecture to enhance Communication-Response-Compute-Security-Storage (CRCSS) in the system. Blockchain technology provides security for the SCM transactions and data. The efficiency of the Blockchain is measured in terms of upload time, download time and transaction fees for Bitcoin, Ethereum and Filecoin platforms. The Filecoin blockchain platform is quicker and cost-effective for larger file sizes, compared to Ethereum and Bitcoin, making it suitable for Pharmaceutical SCM systems.

1. Introduction

Internet of Things (IoT) are used day-to-day, in almost every domain - in homes, industries, healthcare, agriculture, etc. The number of connected IoT devices in 2020 was around 9.76 billion and grew to 13.14 billion by 2022. It is expected that by 2030 there will be more than 29 billion devices connected with each other over the internet [1]. Presently, the IoT has found wide application in Supply Chain Management (SCM). The IoT market for Supply Chain is said to be growing at 15.5% Compound Annual Growth Rate (CAGR) [2]. Supply chain is the transition of goods, services and information from the producer to the consumer. Automating the flow of processes involved in supply chain by using IoT and related technologies is helping to improve the manufacturing turnover

* Corresponding author.
E-mail address: mangala.natampalli@gmail.com (Mangala N.).
[1] Fellow, IEEE.

https://doi.org/10.1016/j.iot.2024.101215
Received 26 February 2024; Received in revised form 23 April 2024; Accepted 6 May 2024
Available online 13 May 2024

and consumer satisfaction [3]. The SCM system comprises of forecasting, sourcing, production, distribution and returns. The IoT network aid in automation of tasks by their ingrained sensors and actuators and ability to exchange data over the Internet.

IoT is used to obtain accurate and timely information about location, condition and status of goods/services in SCM. While nearly all industries and service sectors are using SCM nowadays, there are some critical domains which have stringent demands of timeliness, ordering, accuracy, quality and quantity such as perishable goods, fisheries, pharmaceutical SCM. The creation, manufacturing and distribution of life-saving medications and healthcare items fall within the purview of the pharmaceutical industry. The pharmaceutical supply chain faces several challenges that impact the availability, safety, and efficiency of delivering medicines to patients [4–6]. Big industries such as aeronautical, automobile and semiconductor sectors have complex SCM involving many components, processes and huge amount of information. For example, an airline company SCM has to deal with several thousands of components, their quality, date of installation, chassisID, service due date, etc. and keep track of the operational airplanes to ensure that they are safe. A recent report said that Boeing had advised to ground and inspect all its 737 Max aircrafts when a bolt with a missing nut was discovered in one of its planes [7].

The automated SCM should focus on providing real-time response for the supply chain processes, and ensure security and reliability. The Blockchain technology is very useful for Supply Chain Management Systems. Not only does it automate coordination, it helps in spatio-temporal tracking of components, improve accountability, reduces the risk of fraud, and enhances trust among the entities. Further, in Pharmaceutical SCM systems, blockchain provides additional benefits of preventing counterfeiting, streamlining regulatory compliance, and quick action during emergencies such as drug recall.

The field of Supply chain analysis is gaining importance to enable the business houses to improve planning, sales and operations, inventory management and logistics management. The supply chain data are analyzed to identify and predict future risks by recognizing patterns and trends throughout the supply chain. Breaches in supply chain data stream cause big economic losses. Hence SCM requires process security and data security. While IoT is useful for sensing and controlling the operations in SCM, Blockchain is a shared, immutable ledger for recording transactions, tracking resources and building trust.

A novel IoT-Cloud architecture with intermediary Mist, Edge and Fog layers has been designed to improve the Communication-Response-Compute-Storage. The Blockchain technology plays a dual role of securing the SCM transactions as well as protecting the data storage in Cloud. It is useful for researchers, architects and developers to understand the subtleties of IoT, Cloud and Blockchain technologies for designing sophisticated SCM solutions.

Motivation: Similar works related to Blockchain enabled Pharmaceutical SCM systems, presented in Table 1, are still in concept and design phase.

There are few pre-developed supply chains systems based on blockchain (such as VeChain, IBM Food Trust, OriginTrail, CargoCoin), but they are not integrated with IoT. There are also research efforts involving IoT-Edge-Cloud and Blockchain, by Yu et al. [13], but it is not specifically tested for SCM applications. There are few supply chain frameworks using IoT and Blockchain [14], but it is not optimized for delay-intolerant applications.

Hence, there is a motivation to design a real-time and more secure system for Supply Chain Management and Analytics by suitably adapting the IoT, Cloud Computing and Blockchain technologies.

Contributions: This work demonstrates application of Blockchain security to the IoT-Cloud based pharmaceutical supply chain management to address the problems of confidentiality, integrity, tamper-resistance and timeliness in the manufacture, storage and expiry of drugs. Specific contributions of this work include:

- Presents several real world SCM use case scenarios and the state-of-the-art enabling technologies including Blockchain, IoT, Cloud and AI
- Designing a framework for improving the response time and security of pharmaceutical supply chain requirements using a four layer IoT-Cloud architecture
- Implementation and quantitative analysis of Blockchain platforms — Bitcoin, Ethereum and Filecoin with respect to time delay and file size

Organization: The rest of the paper is organized into following sections. The preliminaries of Supply Chain, Blockchain, and Mist, Edge, Fog and Cloud Computing Paradigms are explained in Section 2. Literature review is summarized in Section 3. The problem statement of Pharmaceutical SCM systems is described in Section 4. The proposed solution using multi-level architecture and blockchain for automating the Pharma SCM is presented in Section 5. A comparison of Bitcoin, Ethereum and Filecoin is presented in Section 6. Implementation procedure for Ethereum and Filecoin platforms is discussed in Section 7. The Experiment details and Performance Analysis is presented in Section 8; including a subsection on Discussions. Conclusion is available in Section 9.

2. Preliminaries

2.1. Supply chain applications

SCM has become an intrinsic part of almost every domain [15]. Nowadays digital technologies like IoT, e-payments, e-commerce are providing automation and speed to the SCM. It is initiated with the procurement of raw materials or components, movement of a product from the producer to the manufacturer, and then it is forwarded to the distributor. The distributor in turn ships it to the wholesaler. Finally, it is available to the customer at the retail stores. Table 2 shows the different phases of supply chain and the stakeholders involved for various industries. Table 3 shows various domains requiring supply chain management. The broad phases

Table 1
Comparison of similar previous works.

Reference	Concept	Achievement	Plausible enhancement
Ghadge et al. [5], 2023, Blockchain implementation in pharmaceutical supply chains: A review and conceptual framework	Provides a conceptual framework of blockchain implementation for pharma supply chain	Architecture of implementation framework of blockchain for PSC	-Implementation of the concept - Use IoT for automation - Extend blockchain for data security also
Abbas et al. [8], 2020, Blockchain and machine learning-based drug supply chain management and recommendation system for smart pharmaceutical industry	(i) blockchain-based drug SCM (ii) ML-based drug recommendation system for consumers	- Manufacturer adds/updates/ deletes drug details in the blockchain and shares with distributors, pharmacies, hospitals, doctors - Customer can query for recommendation of drugs for different disease conditions	- Implement in real-time pharmaceutical companies and finetune performance - Use IoT for automation; Use IoT-Blockchain for automating manufacture process - Securely store data for analytics
Ahmad et al, [9], 2021, A blockchain-based approach for drug traceability in healthcare supply chain	Off-chain Ethereum with smart contracts for immutable history of transactions and product traceability	Protects integrity, availability and non-repudiation of transaction data	- Improve performance by finetuning smart contracts - Extend to manufacturing phase - Automate completely using IoT
Shashank et al. [10], 2020, Blockchain — IoT enabled pharmaceutical supply chain for COVID-19	Proposed Blockchain-IoT pharmaceutical SCM through a literature review	IoT-Blockchain reduce the direct involvement of workers and hence minimize infections	- Efficient implementation of design for real-world time-sensitive use cases
Bapatla et al. [11], 2023, PharmaChain: A blockchain to ensure counterfeit-free pharmaceutical supply chain	Implements a PoC of IoT-Blockchain pharma SCM	Provides tools to verify the authenticity of drugs before consumption	Extend blockchain for secure storage of SCM data in Cloud for analytics and consumer queries
Subramanian et al. [12], 2021, Crypto pharmacy — Digital medicine: A mobile app with hybrid blockchain to tackle issues in pharma supply chain	Every stakeholder has authenticated credentials and can check the status of the medicine through the blockchain	Real-time and temperature monitoring of liquid medicine has been tested with the IoT platform	- Implement for real-world usage - Extend blockchain for secure storage of SCM data in Cloud for stakeholder queries
Our work, secure pharmaceutical supply chain using blockchain in IoT Cloud systems	Four-layer IoT-Cloud architecture with blockchain security for pharmaceutical SCM	- Improved response - Blockchain enabled SCM and secure data storage in Cloud	

of SCM are planning, forecasting, sourcing, production, distribution, and returns. While nearly all industries and service sectors are using SCM, there are some critical domains which have stringent demands of timeliness, ordering, accuracy, quality and quantity. A list of perishable goods and the importance of timelines is presented in Table 4.

2.2. Challenges in the pharmaceutical supply chain

The creation, manufacturing, and distribution of life-saving medications and healthcare items fall within the purview of the pharmaceutical industry, which is essential to the global supply chain. In this industry, it is crucial to guarantee the accuracy, confidentiality, and transparency of supply chain data. Pharmaceutical producers, distributors, pharmacies, and healthcare professionals are just a few of the many stakeholders in the very intricate and widely dispersed pharmaceutical supply chain. Pharma companies require efficient management of a complex supply chain handling challenges related to coordination, communication, procurement, storage, shipping and regulatory compliance. The pharmaceutical supply chain faces several challenges that impact the availability, safety, and efficiency of delivering medicines to patients [4–6]. Some of the main issues in the pharmaceutical supply chain include:

(i) *Counterfeiting and Product Tampering:* Counterfeit drugs pose a significant threat to patient safety. Criminals may produce fake medications that look like legitimate products, putting patients at risk of receiving ineffective or harmful treatments.
(ii) *Cold Chain Management:* Many pharmaceutical products, especially biologics and vaccines, require specific temperature conditions to maintain their efficacy. Ensuring the integrity of the cold chain throughout the supply chain is crucial, and deviations in these parameters can result in product spoilage.
(iii) *Quality Control and Assurance:* Maintaining consistent product quality is critical in the pharmaceutical industry. Quality control processes must be rigorous and continuous to identify and address any deviations in manufacturing that could compromise the safety and efficacy of medicines
(iv) *Demand Forecasting:* Accurate demand forecasting is essential to avoid stockouts or overstock situations. Pharmaceuticals have a limited shelf life, and incorrect forecasting leads to product wastage or shortages, affecting patient access to medications.
(v) *Supply Chain Resilience:* Disruptions, whether caused by natural disasters, geopolitical events, or other factors, impact the supply chain. Developing resilient supply chain strategies, including contingency planning and risk mitigation, is essential to ensure a continuous supply of medicines.

Table 2
Examples of industries/domains requiring supply chain management.

Domain name	Description for the requirement
Perishable goods	Short shelf-life products such as fresh fruits, vegetables, seafood need efficient supply chains to avoid spoilage.
Pharmaceuticals	Medicines/ vaccines require temperature control and timely distribution to ensure they are effective and safe.
Fashion industry	Fast-changing fashion trends require a responsive supply chain to get new styles quickly from design to retail.
Electronics	The components for appliances come from different suppliers and need to be assembled in a coordinated manner.
Automotive industry	Manufacturing cars involves assembling thousands of parts from different suppliers to create a final product.
Agriculture	Seeds, fertilizers, and machinery must be coordinated to ensure a successful harvest.
Retail	Keeping products in stock and ready for customers requires efficient inventory management and distribution.
E-commerce	The rapid growth of online shopping demands effective logistics to deliver products to customers' doorsteps.
Fast food chains	Quick-service restaurants require a streamlined supply of the right ingredients at the right time.
Construction	Materials like steel, cement and lumber need to be delivered to construction sites in a timely manner.
Oil and Gas	Extracting, refining, delivering oil/gas products involves complex logistics.
Mining	Minerals and metals need to be extracted, processed, and transported efficiently.
Healthcare	Hospitals need a steady supply of medical equipment and pharmaceuticals to provide care.
Aerospace	Aviation industry needs precise coordination of parts and materials for aircraft manufacture.
Consumer electronics	Gadgets like smartphones or laptops require coordination of parts from different suppliers.
Beverage industry	Soft drinks, alcoholic beverages, bottled water need to be manufactured and distributed quickly.
Chemicals	Specialty chemicals used in various industries require careful handling and distribution.
Textiles	Fabric production and distribution involve managing multiple suppliers and production processes.
Toys and Games	Timely production and distribution of toys and games are crucial, especially during holiday seasons.
Food delivery services	Platforms like meal kit deliveries require coordination of ingredients and timely delivery.
Energy	Setting up Renewable Energy plants, require transporting equipment/components to installation sites.
Telecommunications	Communication networks require sourcing and delivering many equipment.
Pharma distribution	Distributing prescription drugs to pharmacies and hospitals while adhering to regulations.
Chemical industry	Managing the supply chain for hazardous chemicals requires strict safety measures.
Packaged foods	Processed foods need careful handling to prevent spoilage or contamination.
Home appliances	White goods like refrigerators and washing machines require efficient distribution.
Furniture	Coordinating production and delivery of furniture to retailers and customers.
Luxury goods	High-end designer fashion products require precise management to maintain exclusivity.
Entertainment industry	Physical distribution of CDs, DVDs, and merchandise for artists and media companies.
Clinical trials	Supplying drugs and medical equipment to various trial sites in clinical research.
Food ingredients	Manufacturers sourcing ingredients from multiple suppliers to create recipes.
Automotive aftermarket	Managing spare parts distribution for vehicle repairs and maintenance.
Waste management	Coordinating waste collection, processing, and disposal in a sustainable manner.
Hospitality	Ensuring a consistent supply of linens, toiletries, and other amenities for hotels.
Schools and Institutions	Supplying educational materials, books, and equipment to schools and colleges.
Defense and Military	Coordinating the supply chain for military equipment, uniforms, and supplies.
Chemical fertilizers	Managing the production and distribution of fertilizers for agriculture.
Renovation/Remodeling	Coordinating materials and equipment for construction projects.
Electricity generation	Coordinating the supply chain for power plant equipment and fuel.

(vi) *Product Recalls:* Product recalls are costly and damage the pharmaceutical company's reputation. Identifying and isolating the affected products quickly is crucial, and efficient recall processes are necessary to minimize the impact on patients and the supply chain.

(vii) *Regulatory Compliance:* The pharmaceutical sector is extensively regulated, with strict data management, quality control, and traceability standards. It is critical to follow numerous regulatory standards, such as Good Manufacturing Practices (GMP) and Good Distribution Practices (GDP).

(viii) *Data Security and Cyber Threats:* It is vital to safeguard sensitive patient data, intellectual property, and proprietary information. Data breaches cause serious legal and financial ramifications.

(ix) *Drug Traceability:* Ensuring pharmaceutical product traceability from manufacture to end-user distribution is critical for discovering and recalling potentially dangerous or counterfeit products.

In recent years, supply chain analysis is gaining importance to enable the business houses to improve demand forecasting, inventory management, sales, operations and logistics planning. Supply Chain Analytics is the process of evaluating each stage of a supply chain starting from the time the business acquires raw materials to the delivery of final products to the customers. The data gathered from supply chain can be analyzed to pinpoint known risks and identify future risks by spotting patterns and trends across the supply chain. Hence, it is an important task to protect the supply chain data.

2.3. Blockchain

Blockchain is a distributed and decentralized ledger that is used to record transactions done over multiple computers. The important features of Blockchain are:

- *Immutability:* Blockchain made up of multiple blocks which are connected in a chain structure to maintain a public ledger. Each block incorporates information about the previous block. Once the block (data) is added to the blockchain it is impossible to alter or manipulate the information within it.

Table 3
Stakeholders involved in various phases of supply chain.

Raw material acquisition and production	
Agriculture	Farmers, Suppliers, Distributors
Mining	Miners, Suppliers, Manufacturers
Chemicals	Manufacturers, Suppliers, Distributors
Textiles	Cotton growers, Fabric manufacturers, Clothing brands
Pharmaceuticals	Drug manufacturers, Suppliers, Distributors
Food ingredients	Ingredient producers, Suppliers, Food manufacturers
Chemical fertilizers	Fertilizer manufacturers, Suppliers, Distributors
Manufacturing and production	
Electronics	Component suppliers, Manufacturers, Distributors
Automotive industry	Part suppliers, Car manufacturers, Dealerships
Aerospace	Component suppliers, Aircraft manufacturers, Airlines
Construction	Material suppliers, Construction companies, Clients
Renewable energy	Equipment manufacturers, Energy producers, Consumers
Consumer electronics	Component suppliers, Manufacturers, Retailers
Luxury goods	Artisans, Designers, Retailers
Furniture	Raw material suppliers, Manufacturers, Retailers
Pharmaceutical production	Drug manufacturers, Suppliers, Distributors
Wholesalers and suppliers	
Wholesalers	Wholesalers, Retailers, Consumers
Retail	Retailers, Wholesalers, Consumers
Logistics and distribution	
Transportation	Transport companies, Suppliers, Retailers
E-commerce	E-commerce platforms, Sellers, Customers
Food delivery services	Delivery services, Restaurants, Consumers
Waste management	Waste collection, Processing facilities, Recyclers
Environmental services	Environmental agencies, Waste facilities, Consumers
Electricity generation	Power plants, Suppliers, Consumers
Defense and Military	Defense contractors, Military bases, Governments
Automotive aftermarket	Aftermarket suppliers, Repair shops, Consumers
Renovation and Remodeling	Suppliers, Construction firms, Homeowners
Entertainment industry	Producers, Directors, Actors distributors, Audiences
Consumers and end users	
Customers	Consumers, Retailers, Wholesalers
Schools and Institutions	Educational institutions, Suppliers, Students
Fast food chains	Chain restaurants, Suppliers, Customers
Fashion industry	Designers, Manufacturers, Consumers
Beverage industry	Beverage manufacturers, Distributors, Consumers
Packaged foods	Food manufacturers, Distributors, Consumers
Hospitality	Hotels, Suppliers, Guests
Health care	Medical facilities, Pharmaceutical distributors, Patients

- *Time-stamping:* The time of transaction is recorded in the block.
- *Traceability:* Similarly, the location of the goods along with time are recorded in the blockchain.
- *Transparency:* All transactions are transparently viewable by all participants.
- *Automation:* The smart contracts are code snippets stored on a blockchain that execute when predetermined conditions are met; they are used to automate the execution of agreements among participants.

These features help to address the challenges of SCM. For example, drug traceability is taken care due to time-stamping and location traceability; Regulatory Compliance is achieved due to the transparency feature. The blockchain is largely used to store and record transactions of cryptocurrencies.

There are many areas where blockchain technology is used, such as:

- *Payments and Money Transfers*: Millions of users are transferring money over the blockchain which is a very hassle-free way to transfer or do the payments. We can also avoid bank transaction fees as well and there is no limit on how much money is transferred in a day or by a single transaction, unlike traditional banks. This means millions and billions of dollars can be transacted with just a touch and within a few seconds.
- *Monitor supply chains*: Blockchain is used to monitor supply chain business as the technology operates in real-time, it is easy to track the supply and manufacturing of all the other stuff as the transactions are stored and are viewed whenever required. Also, once the data is entered on the blockchain it cannot be changed, it acts as a security as well.
- *Data Sharing*: This technology is useful to store and transfer data securely, especially in big industries while moving or transferring data which are confidential.

Table 4
Perishable products and their SCM requirements.

Product name	Description
Fresh flowers	Flowers have a short lifespan once cut and require fast distribution to maintain freshness.
Bakery goods	Bread, pastries and other baked foods have limited shelf lives and need to be delivered quickly to retailers.
Seafood	Fresh seafood like fish and shellfish spoil rapidly, necessitating rapid distribution.
Medical supplies	Medicines, blood and medical items mandate fast and precise delivery.
Live plants	Nursery plants need to be transported quickly to prevent dehydration and damage.
Live seafood	Live seafood like lobsters and crabs should be delivered promptly to maintain their quality.
Fresh herbs	Short shelf life Culinary Herbs need to be quickly transported to markets/ stores.
Dairy products	Milk, yogurt, fresh cream have limited shelf lives and need timely distribution.
Prepared meals	Ready-to-eat meals and salads have short expiration times and require efficient logistics.
Sushi	Sushi and other raw fish dishes need fast delivery to maintain their taste and quality.
Cut fruits	Freshly cut fruits sold in grocery stores or restaurants require rapid distribution.
Ice cream	Frozen treats like ice cream and gelato need careful temperature-controlled distribution.
Fresh juice	Cold-pressed juices have a short window for consumption and need quick delivery.
Specialty coffee	Freshly roasted coffee beans have a limited freshness period; should be distributed promptly.
Live poultry	Live chickens or other poultry need to be delivered quickly to processing facilities.
Limited-edition foods	Items like holiday-themed chocolates require fast supply chains to meet demand and capitalize on trends.
Exotic fruits	Fruits that are delicate and uncommon require quick distribution to maintain quality.
Fresh seafood	Like live seafood, fresh fish and other seafood need fast delivery to prevent spoilage.
Temperature-sensitive medications	Some medications require strict temperature control and fast distribution.
Microbrewery beer	Crafted beer have short shelf life and requires efficient distribution.
Organic produce	Organic produce may have shorter shelf lives due to the absence of preservatives.
Kombucha	This fermented tea drink has limited shelf life and requires careful distribution.
Salads	Fresh salads and delicate greens, need to be delivered quickly to maintain crispness.
Raw meats	Raw meats have short freshness periods and need fast distribution.
Limited-time promotions	Special edition foods or drinks require quick supply chain management to take advantage of short-term demand.
Eggs	Fresh eggs need timely delivery to retailers due to their relatively short shelf life.
Fresh spices	Some fresh spices have limited shelf lives and require efficient logistics.
Seasonal fruits	Fruits that are available only during specific seasons need fast distribution to maximize sales.
High-end desserts	Luxury desserts like macaron require quick delivery to maintain their quality.
Fresh meats	Apart from raw meats, marinated or seasoned meats have short freshness periods.
Fresh oysters	Oysters and other shellfish should be delivered quickly to ensure quality and safety.
Sliced deli meats	Deli meats like ham and turkey need fast distribution to prevent spoilage.
Potted plants	Potted plants and flowers require quick delivery to maintain their appearance.
Exotic meats	Uncommon meats like game meats have short shelf lives and require efficient distribution.
Fragile desserts	Delicate desserts like souffle need careful handling and rapid delivery to prevent collapse.
Fresh pasta	Freshly made pasta has a short freshness period and requires timely distribution.
Artisanal cheeses	Specialty cheeses often have short shelf lives and need to be delivered promptly.
Fresh squeezed juices	Natural juices need to be delivered fast to maintain their nutritional value.
Expiring cosmetics	Cosmetics containing natural ingredients require efficient supply chains.
Live insects	Insects used as food for pets require fast and careful delivery.
Fresh sausages	Uncooked sausages need prompt delivery to maintain their quality and taste.
High-protein snacks	Protein bars and energy snacks need to be delivered quickly to maintain their freshness.
Fresh sauces	Sauces with no preservatives have limited shelf lives and need efficient logistics.
Nutritional supplements	Supplements with natural ingredients may have short expiry times.
Farm-to-table	Locally sourced produce needs fast delivery to maintain its farm-fresh quality.
Chilled soups	Freshly prepared chilled soups need quick distribution to preserve taste and texture.
Small-batch jams	Handmade jams often have limited shelf lives and need quick distribution.
Fresh baby food	Natural baby food products require efficient logistics to ensure they are safe for consumption.

- *Copyright and Royalties Protection*: Blockchain is used for transparent tracking of art/music of artists and provide them real-time royalties.
- *Healthcare*: It is used to maintain and track the medical health records and clinical trials of patients. As the data once entered cannot be modified, it also provides proper security to the medical records.

2.3.1. Blockchain architecture

The architecture of Blockchain is depicted in Fig. 1. It consists of multiple blocks connected in a chain-like structure. Over a set of transactions, new blocks are created, and connected with each other. Every new block contains the previous block's hash and every block has a unique hash. Each transaction contains the cryptographic hash, transaction date, transaction time and transaction data, and hence the blockchain is tamper-proof.

The main key concepts of blockchain are:

- *Distributed ledger*: A distributed ledger is a dispersed system of records that are shared across a network. In a distributed ledger, the transactions are recorded only once, which eliminates duplication on the networks.
- *Permissions*: Permissions ensure that transactions are secure, authenticated and verifiable. Organizations can easily comply with data protection standards, such as those mandated in the Health Insurance Portability and Accountability Act (HIPAA) and the EU General Data Protection Regulation (GDPR).

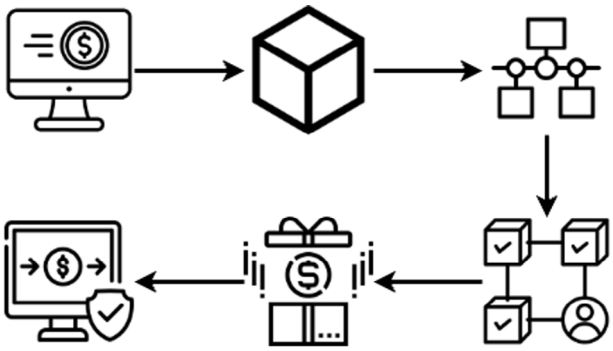


Fig. 1. Working of blockchain.

- *Smart contracts*: A smart contract is defined as an agreement or set of rules that governs a business transaction; it is stored on the blockchain and automatically executed as part of the transaction
- *Consensus*: Through consensus, all parties agree to a network-verified transaction. Blockchains have various consensus mechanisms such as *Proof-of-Work* (PoW), *Proof-of-Authority* (PoA), *Proof-of-Stake* (PoS), *Multi-Signature and Practical Byzantine Fault Tolerance* (PBFT) [16]. This is the core part of the blockchain technology and generally compute intensive.

2.3.2. Popular blockchain platforms

The blockchain technology originally implemented with immutability and has overtime evolved to include many features such as smart contracts, Decentralized Applications (DApps), scalability, interoperability, efficiency etc. There are many popular Blockchain platforms — Bitcoin, Ethereum, FileCoin, Hyperledger Fabric, Corda, IOTA, VeChain, Tron, Stellar, NEO, MultiChain, EOS, OpenChain, Quorum, NEMA and many more. Each blockchain platform has a different set of capabilities suitable for different applications' needs. The important aspects to be considered while choosing a Blockchain for an application are:

(i) *Type:* It can be Public, Private, Hybrid or Consortium of participating nodes.
(ii) *Consensus Mechanism:* The algorithms such as PoW, PoS, PoA, or PBFT which is the core of the blockchain, determine the time consumed to arrive at consensus in the blockchain.
(iii) *Latency:* The upload time or transaction confirmation time is related to the complexity of consensus algorithm, number of nodes participating in consensus, block size, compute nodes and network bandwidth.
(iv) *Size:* Blockchains have size limits for data payload being uploaded and the size per block in on-chain mode for simple transactions. For example, Bitcoin has a 1 MB Block size, Hyperledger Fabric has 100 MB per payload and IOTA supports upto 32 kB. To accommodate higher data sizes blockchains operate in off-chain mode in which the large files are stored in Cloud and it is transaction metadata is stored in the blocks.
(v) *Transaction Fees:* The Blockchain platform may be free or paid. To use Bitcoin, the sender must pay transaction fees; while Hyperledger Fabric is a free blockchain.
(vi) *Transaction Privacy:* Determines if the transaction details are public. Usually the transaction details are public (Bitcoin, Ethereum, Filecoin, IOTA, VeChain) but few blockchains provide good privacy of the transactions (Hyperledger Fabric, Corda)

The users choose the blockchain platforms, based on the above factors. Blockchain platforms are also optimized for application domains, such as Bitcoin and Ethereum are targeted for Decentralized Finance (DeFi), while IOTA and VeChain are more suited for SCM. This work investigates the Bitcoin, Ethereum as well as Filecoin blockchain platforms.

2.3.3. Procedures in blockchain

Some of the important procedures used in blockchains are explained in this subsection.

A. Generate SHA-256 Hash

Function 1: Generate SHA-256 Hash
Input: Data; **Output:** Computed hash

Initialize SHA-256 hash object
Update hash object with Compute hash
return computed hash

Initialize SHA-256 Hash Object: To begin, initialize a SHA-256 hash object. This object is provided by a cryptographic library for SHA-256 hashing algorithm.

Update Hash Object with Data: Once the hash object is initialized, update it with the input data required to be hashed. This data can be any sequence of bytes, such as a string, file contents, or binary data.
Compute Hash: After updating the hash object with the input data, the cryptography functions provided by the library to compute the actual SHA-256 hash is invoked. The library takes care of performing the necessary mathematical operations to transform the input data into a fixed-length hash value.
Return the Computed Hash: Finally, the computed SHA-256 hash is returned as the output of the algorithm. This hash is a fixed-size string of characters that uniquely represents the input data. Even a small change in the input data would lead to a significantly different hash.

B. Verify Transaction

Function 2: Verify Transaction
Input: transaction, previous_ledger_state; **Output:** verification result

```
if transaction is well-formed then
    if transaction inputs are valid then
        if transaction is not a double spend then
            if transaction signatures are valid then
                if transaction outputs are valid then
                    if inputs match previous ledger state then
                        update ledger state with outputs  return 'Transaction is valid'
                    end
                    else
                        return 'Invalid: Inputs do not match ledger state'
                    end
                end
                else
                    return 'Invalid: Transaction outputs are invalid'
                end
            end
            else
                return 'Invalid: Transaction signatures are invalid'
            end
        end
        else
            return 'Invalid: Double spending detected'
        end
    end
    else
        return 'Invalid: Transaction inputs are not valid'
    end
end
else
    return 'Invalid: Malformed transaction'
end
```

Transaction Well-Formedness: The first step is to check if the transaction follows the expected structure and format. This involves verifying that the transaction has the required fields and adheres to the correct format.
Transaction Input Validation: Ensure that the inputs referenced in the transaction are valid and exist in the ledger. This step involves checking that the referenced inputs have not been spent before.
Double Spend Detection: Ensure that the inputs referenced in the transaction are not already spent in other transactions. This prevents double spending of the same funds.
Transaction Signature Verification: Verify the digital signatures associated with the transaction. This step ensures that the transaction was signed by the rightful owner of the funds.
Transaction Output Validation: Check that the outputs of the transaction adhere to the defined rules of the blockchain, such as the correct format and values.
Matching Inputs and Ledger State: Ensure that the inputs specified in the transaction match the current state of the ledger. This step confirms that the funds being spent belong to the sender.
Update Ledger State: If all the checks pass, update the ledger state by adding the new outputs of the transaction. This step ensures that the blockchain accurately reflects the transfer of ownership.

C. Proof of Work

Function 3: Proof of Work
Input: block_data, target_difficulty; **Output:** nonce, calculated_hash

$nonce \leftarrow 0$ **while** ***true*** **do**
  $combined\_data \leftarrow$ concatenate($block\_data, nonce$) $calculated\_hash \leftarrow$ hash_function($combined\_data$) **if** $has\_required\_prefix(calculated\_hash, target\_difficulty)$ **then**
    return $nonce, calculated\_hash$
  end
  $nonce \leftarrow nonce + 1$
end

Parameters: The input parameters are, *(i) block_data*: Data of the block being mined and *(ii) target_difficulty*: Number of leading zeros the hash should have.
Initialization: Initialize a variable nonce to 0. Miners alter the nonce to affect the hash.
Loop: Enter an infinite loop using while true.
Hash Data: Create data_to_hash by concatenating block_data and the current value of nonce.
Hash Calculation: Compute hash_result by applying a hash function (e.g., SHA-256) to data_to_hash.

D. Check Required Prefix

Function 4: Check Required Prefix **Input:** hash_result, prefix_lenght; **Output:** boolean result

$required\_prefix \leftarrow$ "0" $\times\, prefix\_length$
return hash_result.startswith(required_prefix)

Check Difficulty: Utilize the has_required_prefix function to check if hash_result satisfies the specified target_difficulty.
Condition Met: If the condition is met (hash has required prefix), return nonce and hash_result.
Condition Not Met: If the condition is not met, increment nonce and the loop continues.

2.4. Inter-planetary file system

The Inter-Planetary File System(IPFS) is an open-source file-sharing peer-to-peer network used to store and share data in a distributed file system [17]. IPFS uses content-addressing to uniquely identify each file in a global namespace connecting IPFS servers. Similar to BitTorrent, IPFS enables users to host and receive content. A resilient system of file storage and sharing is created by IPFS, which is structured around a decentralized system of user-operators who each hold a piece of the total data. Using a distributed hash table (DHT), other peers in the network can find and request the content from any node that has it, and any user in the network can serve a file by its content address.

The steps involved in IPFS (depicted in Fig. 2) are as follows:

(i) The process of Data Sharing on IPFS begins with first placing/uploading the File into the IPFS.
(ii) Once the uploading of the data is done, it returns a Hash Key along with a timestamp which consists of day, date, and time.
(iii) The user then Queries the Smart Contract for the Public Key of the Worker.
(iv) Further, the file is split into n shares and given keys for each share randomly for encryption purposes.
(v) Finally the Encrypted Shares are stored on the Blockchain.

The goal of IPFS is to establish a single worldwide network, in contrast to BitTorrent. The peers downloading the content from *user 1* and the peers downloading it from *user 2* swap data if two users publish a block of data with the same hash. By using gateways that are HTTP-accessible, IPFS aspires to replace the technologies used for delivering static web pages. Users have the option to use a public gateway rather than installing an IPFS client on their device. On the IPFS GitHub website, a list of these gateways is kept up to date.

2.4.1. Key concepts of IPFS

- *Content Addressing*: IPFS uses content addressing, as opposed to the conventional web, which locates information using URLs depending on the location of the server. Based on its content, each piece of data is given a distinct cryptographic hash, ensuring that pieces of data with the same content always have the same hash. This hash acts as the content's address, making retrieval quick and safe.

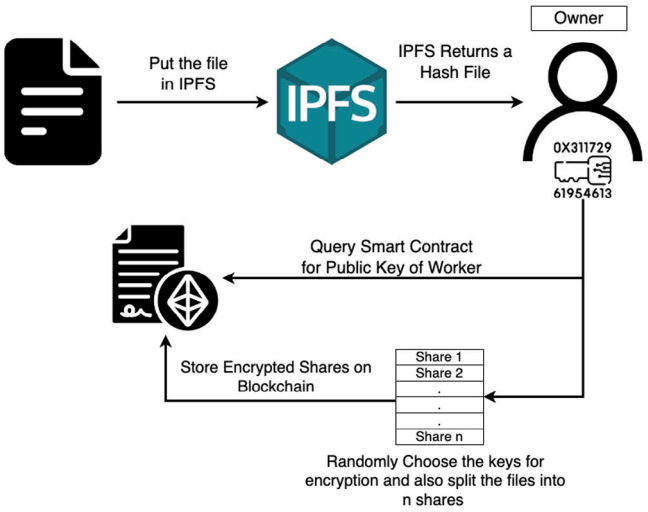


Fig. 2. Data sharing on IPFS.

- *Decentralization*: IPFS does away with data storage and distribution being centralized. In a peer-to-peer network, files are dispersed among a collection of linked nodes. Files can be retrieved from any node that possesses the content, each of which stores a portion of the total material.
- *Merkle DAG*: To represent the connections between pieces of content, IPFS creates a Merkle Directed Acyclic Graph (DAG). A piece of content is represented by each node in the graph, and edges connect nodes based on content references. Effective versioning, data integrity, and content sharing are all made possible by this structure.
- *Data Deduplication*: IPFS uses data deduplication because identical content is given the same hash. This reduces storage redundancy and optimizes network traffic because a file that is uploaded by several users is only stored once on the network.
- *Cryptographic Integrity*: Content addressing protects cryptographic file integrity, which is a requirement for encryption. Since the material is uniquely recognized by its hash, any modifications cause a change in the hash, making tampering or unauthorized adjustments immediately obvious.
- *Rewards and Incentives*: The IPFS network offers the idea of *Filecoin*, a cryptocurrency that runs on top of the IPFS network. By offering material to people who seek it and sharing their storage space with the network, users can earn Filecoin. This encourages involvement and aids in keeping the content accessible.

2.4.2. Benefits of IPFS

- *Decentralization*: By reducing reliance on centralized servers, IPFS improves the availability and resilience of data. It lowers the chance of censorship and single point failures.
- *Faster material Retrieval*: Since the system can locate material based on its hash, content addressing enables efficient retrieval, allowing for faster downloads by collecting data from close-by nodes.
- *Offline Access*: Because IPFS nodes include an offline mode, users can access content even when they are not connected to the internet. This is especially helpful in locations that are far away or disaster-stricken.
- *Data preservation*: As long as someone on the network keeps the content, its permanence is improved. For long-term archiving and preservation of data, this is advantageous.
- *Reduced Bandwidth*: Because IPFS makes use of data deduplication, less bandwidth is needed to transfer similar material.

2.5. Computing paradigms

IoT depends on external computing paradigms such as Cloud computing for processing and analytics. This work leverages the four layered Mist-Edge-Fog-Cloud architecture being patented by Venugopal et al. 2024, *An Integrated Computing Storage System and Method of Security related thereto*, [18], to enhance Communication-Response-Compute-Security-Storage (CRCSS) performance for a Pharmaceutical SCM.

(i) *Mist Computing*: Mist Computing, is a distributed computing paradigm that brings cloud computing closer to the data source or edge devices. It tries to enhance efficiency and minimize latency by processing data locally, close to where it is generated, rather than transferring all data to a centralized cloud server. Communication takes nearly five times the power of computing in embedded microcontroller, so, by collecting and processing raw data at the very edge conserves time, bandwidth and battery power. For example, Mist computing can be used to filter information at the source level so as to reduce data transmission and storage requirements and speed up future data analysis [19]. Filtering, anomaly identification mechanisms,

geospatial calculations or pattern recognition are performed on the mist. The time delay in Mist Computing is negligible; but the processing capability is very limited [20,21]. Fiber optic networks provide lossless and fast communication [22,23], but the Wireless Sensor Networks are popular in IoT communication [24,25].

(ii) Edge Computing: The Edge Computing (EC) architecture emerged in the early 2000s, as a low capacity intermediate layer between the IoT and Cloud to address the problem of transmission delay. The edge nodes provide computing offload, storing and caching for IoT management for less computation and fast response jobs [26]. Edge computing seeks to improve response times, reduce network congestion, and increase privacy by processing data locally before transferring it to the cloud for processing [27]. Hence, Edge computing nodes directly connect to the cloud as well as the sensors, but generally do not communicate laterally with other edge nodes and can only work in a single application per setup. Data generated by the sensors are forwarded to the edge nodes, where nominal processing is carried out on the data and fast response to queries are provided back to the respective query device [28]. The difference between Edge and Fog is that the Edge node is closer to the source IoT layer, has relatively lower computational capacity, but also has a low data transmission time due to its proximity to the device, and rarely require the processing capability of the cloud.

(iii) Fog Computing: Fog Computing brings the computing capability in between the IoT devices and the Cloud server, to address the increased network traffic and latency caused by cloud computing. Fog is a medium weight and intermediate level of computing power which complements Cloud Computing [29]. The advantage of Fog computing is that it reduces response time; it also reduces the amount of data that needs to be sent to the cloud. Rather than forward all the IoT data to Cloud servers, some processing is done on fog nodes for faster response, and processed data (fit for archival or long term storage) is stored in Cloud, and bandwidth requirements are reduced.

(iv) Cloud Computing: Cloud Computing (CC) is the on-demand network access to a shared pool of configurable computing resources (compute servers, storage, applications, databases) [30]. Cloud data centers are typically huge facilities set up by big organizations. The Cloud APIs facilitate provisioning of resources, deployment of VM images, storing of data, and help to enable the IaaS/PaaS/SaaS service delivery models in Cloud Computing [31]. The on-demand dynamic elasticity, geographically independent network access and elimination of cost of ownership are the key advantages of Cloud; which are useful for IoT data processing and storage. Theoretically, the Cloud is of unprecedented compute-storage capacity; and the IoT connects to Cloud Computing over WAN or through cellular networks. However, the time delay in transmission of data between IoT and Cloud, over the Internet, is problematic for time-critical systems. This issue has been addressed in Fog Computing [32].

2.6. Machine learning and artificial intelligence

Machine Learning (ML) is a type of Artificial Intelligence (AI) that is based on Artificial Neural Network (ANN), especially Convolutional Neural Networks (CNN), which includes multiple layers of processing used to extract high levels of features from data [33]. This approach allows for the creation of sophisticated models that can recognize patterns and make predictions in a wide range of applications, from speech recognition and image processing to natural language understanding and autonomous driving. The term *deep* refers to the depth of the neural network, which can have many layers, each of which processes different aspects of the data. By training these networks on vast amounts of labeled data, deep learning algorithms can learn to recognize subtle patterns and make accurate predictions with remarkable accuracy. As such, deep learning is becoming increasingly important in fields such as healthcare, finance, marketing and SCM for analytics of large data sets, forensics and complex decision-making processes.

3. Literature review

Ghadge et al. [34], address the need of blockchain technology in the pharmaceutical supply chain, particularly in light of the Covid-19 pandemic. The proposed conceptual framework offers practical guidance for stakeholders to navigate the implementation of blockchain, fostering transparency, traceability, and efficiency in the pharmaceutical supply chain. Singh et al. [35], implemented a blockchain-enabled patient EHR healthcare framework with smart contracts to manage multistakeholders (doctors, patient, pathology, chemist, insurance) while guaranteeing data privacy, availability, immutability, authentication. The demonstration indicates that blockchain-enabled EHR frameworks will revolutionize next-generation healthcare.

Qun et al. [14], present a comprehensive framework for supply chain management based on the Internet of Things (IoT) and blockchain technology. The framework consists of three main components: an access control system, a backup peer mechanism, and an internal data isolation and transmission approach. The access control system includes a registrar module and an inspection module. The registrar module handles information registration with a predefined policy that all companies in the supply chain must adhere to. It also provides functions for revocation and updating. The inspection module monitors the actions of the subjects and identifies any misbehavior, allowing the system to penalize violators. By verifying and storing all actions and information in the blockchain, the proposed framework enhances IoT access control and the security of IoT admission. The proposed solution enhances security, resource allocation, and waste management while promoting effective collaboration among companies in the supply chain. A supply chain system for IoT using hyper-ledger blockchain fabric with efficient data isolation and transmission and improved security has been designed. Integration of the existing system and its cost-effectiveness needs to be further examined.

Mozhdeh et al. [29], have designed a blockchain-enabled fog structure to provide secure IoT with reduced latency for real-time data processing. The issue of scale-up in maintaining high performance and energy efficiency needs to be further examined. Mengmeng et al. [36], have developed a location privacy-preserving cloud-sensing system using smart contracts in a private

blockchain with an authentication algorithm. It protects the privacy of the worker's location and increases the success rate of the completion assigned and helps prevent attacks through re-identification.

However, the system's resilience to sensitive data needs improvement.

The worker's acceptance probability p_w is modeled as a function of both distance d_w,t and reward R_t^w, as given below:

$$p_w = f(d_{w,t}, R_t^w) = \begin{cases} y(\alpha \frac{R_w^t}{d_{w,t}}), & d_{w,t} \leq max D \\ 0, & d_{w,t} > max D \end{cases} \quad (1)$$

Kumar et al. [37], have used Random Forest, Xtreme Gradient Boosting, Message Queuing Telemetry Transport (MQTT) and BoT-Dataset, to effectively detect the IoT-based attacks. The Blockchain provides a decentralized network. Further, it can be extended by adding deep-learning techniques and improving the performance and the latency time to detect the attacks.

The value $\epsilon = 0.1$ is set to the parameters so that overfitting can be avoided. The parameter value is predefined, which causes degradation in the value of the prediction.

$$v_k^{(s)} = v_k^{(s-1)} + \in f_s(r_k) \quad (2)$$

Yu et al. [13], proposes a hierarchical edge-cloud blockchain called LayerChain for large-scale low-delay Industrial IoT applications. It distributes the blockchain between Edge and Cloud layers to mitigate delays and improve scalability. This approach improves the blockchain with respect to delay, block propagation time and resource requirements. Okafor et al. in their several works propose different methods to optimize the data transfer latency between IoT and Cloud, to improve the response of the IoT-Cloud applications. A method of processing data on the egde without first transferring the data to Cloud is proposed in [38]; a spine-leaf Fog computing network (SL-FCN) is used to reduce latency and network congestion in the distributed multilayer IoT-Fog-Cloud. A CPS architecture named CloudMesh to support Input–Output (IO) data stream, traffic engineering and software defined network monitoring in Edge-to-Fog and Fog-to-Cloud is discussed in [39]. An optimized data transmission using Software-defined network optimization (StreamRobot) for IoT networks is proposed in [40]. A lightweight multi-hop routing protocol performs optimal routing in a multilayered system to improve throughput and signal stability while reducing path loss and communication cost has been attempted [41].

The summary of recent literature reviewed is presented in Table 5.

4. Problem statement

4.1. Pharmaceutical Supply Chain Management System (PSCMS)

The Pharmaceutical Supply Chain comprises of purchasing raw materials to manufacture of medicines and delivering it to the consumers. The processes and mobility of data in Pharma SCM are as follows:

(i) *Procuring raw materials:* Verify authenticity and quality from suppliers; Record details on blockchain (eg: batch number, supplier info)
(ii) *Manufacturing:* Record production milestones on blockchain
(iii) *Quality control and testing:* Verify product authenticity and quality; Update blockchain with test results and certifications
(iv) *Packaging and labeling:* Record packaging information on blockchain
(v) *Distribution to wholesalers/retailers:* Track shipment status on blockchain; Handle delivery and payment terms
(vi) *Retail sales:* Record sales transactions on blockchain; Manage payments and inventory updates
(vii) *End customer:* Access product information (eg: authenticity, origin) *via* blockchain; Provide feedback or report issues on the blockchain

4.2. Requirements of PSCMS

The Pharmaceutical supply chains dealing with life-saving medications, require secure manufacturing and storage and the timely delivery of medicines to patients.

(i) PSCMS system should be secure
(ii) It should alert any problems such as tampering, traceability, etc.
(iii) PSCMS systems require fast response
(iv) The system should be scalable
(v) The SCM systems are becoming complex in terms of number of transactions and data
(vi) Some of the large SCM data need to be securely stored and analyzed
(vii) Should be able to deal with complex growing data

Table 5
Blockchain literature review.

Author concept/Model	Algorithm/ Implementation	Performance/ Advantages	Research gaps/ Future challenges
Yuan et al. 2022 [42] Semi-centralized trust management based on blockchain for data exchange in IoT system	- Rotation-based consensus protocol	- Decentralized and semi-centralized architecture - Improved trust computation - Identification of malicious devices	- Scalability - Robustness
Tan et al. 2021 [43], Blockchain-empowered access control framework for smart devices in green IoT	- W3C consortium - Decentralized identifiers (DIDs) - Authentication algorithm - Smart contract in private blockchain	- Unified GSD management platform - Efficient secured model - Protection of text-based data	- Interoperability with GSDs and GIoT platforms - Cost-effectiveness
Xingjuan et al. 2021, [44] Sharding scheme-based many-objective optimization algorithm to enhancing security in blockchain-enabled IIoT	- Main framework of MaOEA-DRP	- Dynamic reward and penalty mechanism - Throughput improvement - Better performance - Security and privacy preservation	- Improve malicious node detection - Reduce energy consumption - Efficient privacy-preserving methods
Qun et al. 2021 [14], Supply-chain system framework based on IoT using blockchain technology	- Committing transactions on the hyperledger fabric	- Efficient data isolation and transmission - Fault tolerance - Improved security	- Integration with existing systems - Cost-effectiveness
Mengmeng et al. 2021 [36], A Blockchain-based location privacy-preserving crowdsensing system	- Register in the public blockchain - Task assignment - Contract creation - Authentication algorithm - Smart contract in private blockchain	- Privacy of worker locations - Better success rate of completing assigned tasks - Prevent re-identification attacks	- Evaluate the proposed system's resilience to attacks - Protect other types of sensitive data
Ismaeel et al. 2021, [45], Incentive-based mechanism for volunteer computing using blockchain	- Blockchain formation using the complex search procedure	- Reduced delays - High reward distribution - Fair and balanced resource usage - Secure comm. & service delivery	- Improve fault tolerance - Improve interoperability
Kebira et al. 2022, [46], Access control and privacy-preserving blockchain-based system for diseases management	- Register or Update patient - Register new device - Add permission - Send data hash & Get data hash - Inter-planetary file system - Proxy re-encryption	- Faster data storage due to Ethereum blockchain - Remote patient monitoring - Scalability due to use of IPFS - Enhanced privacy and security	- Integration with AI - Data standardization - Improve performance
Jiawei et al. 2022, [47], Efficient blockchain hierarchical data sharing in healthcare IoT	- Blockchain-based Hierarchical data sharing framework (BHDSF)	- Fine-grained access control - Efficient retrieval - Aggregative authentication - Secure trustworthy metadata	- Integration with AI - Interoperability
Jiang et al. 2022, [48], Attribute-based encryption with blockchain protection scheme for electronic health records	- Attribute-based encryption (ABE) - CEC-ABE - CP-ABE	- Better patient control over EHRs - Protect against unauthorized modification - Fine-grained attribute revocation - Computational efficiency	- Enhance data privacy and security - Scale up to handle large amounts of data
Mozhdeh et al. 2018, [29], Blockchain-enabled fog structure to provide data security in IoT applications	- Fog computing - DLT	- Improve data security - Reduce latency - Real-time data processing	- Improve energy-efficiency - Scale-up while maintaining good performance - Develop standardized protocols and interfaces
Yu et al. 2021, [49], Blockchain-enhanced data sharing with traceable and direct revocation in IIoT	- Decisional Bilinear Diffie–Hellman (DBDH) - Smart Factory Big Data (SFBD)	- DBDH assumption and ABE ensure security & confidentiality of SFBD - Public/private keys are smaller compared to other schemes - Less overhead time	- Enhance data privacy and security - Cost-effectiveness - Efficient and Compatible

(*continued on next page*)

Table 5 (*continued*).

Author concept/Model	Algorithm/ Implementation	Performance/ Advantages	Research gaps/ Future challenges
Zawar et al. 2022, [50], Blockchain based solutions to mitigate Distributed Denial of Service (DDoS) Attacks in the internet of things: A survey	- Proof of Work (PoW) - Proof of Stake (PoS) - Delegated Proof of Stake (DPoS) - Elliptic Curve Digital Signature Algorithm (ECDSA)	- Difficult for attackers to tamper with data - Difficult for attackers to launch DDoS attacks due to decentralization - Resilience	- Reduce energy-consumption - Privacy-preserving blockchain solutions
Gunasekaran et al. 2022, [51], Blockchain assisted secure data sharing model for internet of things based smart industries	- Blockchain-assisted secure data sharing (BSDS) - Robust Certificate-Less Signature (RCLS)	- Reduced failure rate - Increased response rat - Scalability due to BSDS model	- Integration with legacy systems - Implement in IIoT systems with limited resources
Zhang et al. 2019, [52], Chronos+: An accurate blockchain-based time-stamping for cloud storage	- Window of Time-Stamping (WoT) - BLS signature algorithm	- High accuracy - Tamper-proof - Efficient in terms of monetary costs, WoT and computation	- Compatibility with different blockchains - Cloud-storage
Hewa et al. 2022, [53], Fog computing and blockchain-based security service architecture for 5G industrial IoT-enabled cloud manufacturing	- Certificate generation smart contract - Fog blockchain node requesting verification of smart contract - CSP request verification and Key establishment smart contract	- Reduced end-to-end latency - Increased overall system performance - Improved security and privacy of cloud manufacturing process and equipment	- Integration with Cyber-Physical Systems (CPS) - Integrate with local 5G operators for real-time health monitoring
Yu et al. 2021, [13], LayerChain: A hierarchical edge-cloud blockchain for large-scale low-delay IIoT	- Clouds and edge layered structure to distribute blockchain data - Tree-based clustering algorithm to propagate blocks	- Improved response time - Improved scalability	All edge nodes may not have sufficient resources
Singh et al. 2020, [35], A novel patient-centric architectural framework for blockchain-enabled healthcare applications	- Patient-centric blockchain enabled framework - Stakeholders transactions validation	Throughput (TPS) increased by increasing block size	Include byzantine fault tolerance ordering services for fault tolerance

IoT is an obvious technology for automation in real-world SCM applications; while Blockchain facilitates the process and data security. Since Blockchain offers immutability, time-stamping, traceability and non-repudiation, it is capable of alerting problems such as tampering, counterfeiting and time-delay. The comprehensive set of security properties essential for smart applications include — Confidentiality, Integrity, Availability, Privacy and Non-repudiation (CIAPNr). Blockchain inherently guarantees integrity, availability and non-repudiation. Confidentiality and privacy are typically safeguarded through Cryptography.

In recent years, light weight and homomorphic encryption techniques are being researched to secure resource constrained IoT and for protecting confidentiality and privacy during analytics [54]. The state-of-the-art light weight and hardware cryptographic techniques suitable for resource constrained IoTs are addressed in [55–58]. Kermani et al. [59], have carried out extensive work on fault detection in VLSI with regard to overhead costs, space complexity and time delay. A security module for implantable and wearable medical IoT devices has been proposed [60,61]. Recently, cryptographic techniques are becoming inadequate due to emergence of quantum computers which can easily generate prime factor keys and side-channel attacks. Canto et al. [62] and Koziel et al. [63] have proposed robust quantum safe cryptographic techniques to safeguard against such attacks. Proven solutions for fault detection, countermeasures to attacks and quantum safe cryptography are presented in [64–66]. For performing efficient data analytics, it is recommended to use Homomorphic Encryption, Searchable Encryption and Multiparty Computation techniques, which permit analytical operations on the data aggregation without decrypting each record. A light weight, quantum secure and homomorphic encryption suitable for IoT-Cloud environment has been designed and developed in [54].

Besides enhanced response time and security, other challenges of SCM are due to large number of IoTs and processes. For example, the airplane manufacturing SCM has to deal with several thousands of components, their quality, date of fixture or service due date, and keep track of the operational airplanes to ensure that they are safe. Another challenge faced in the perishable items SCM, is that the quality of ingredients changes with time.

4.3. Objectives

The objective of this work is to determine the most suitable open-source blockchain platform for real-world applications which have requirement of fast response, scalability, data storage, security and efficiency. The objectives of this work are as follows:

- Design an automatic IoT-Cloud-Blockchain based Pharmaceutical SCM with enhanced Communication-Response-Compute-Security-Storage.
- Identify appropriate blockchain platform for PSCMS based on quantitative analysis of upload time, download time and transaction fees

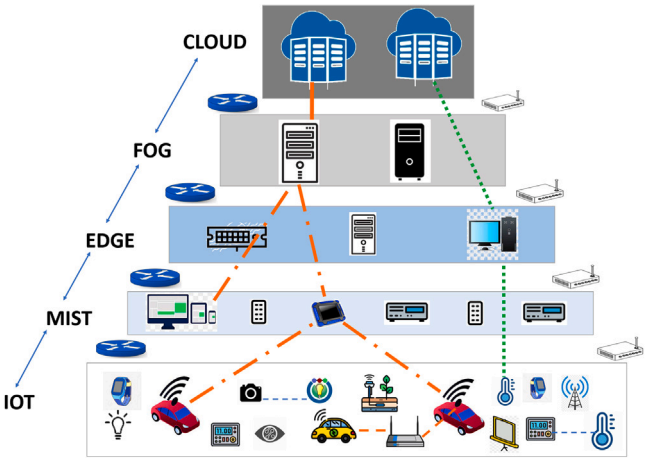


Fig. 3. Architecture of blockchain enabled pharma SCM.

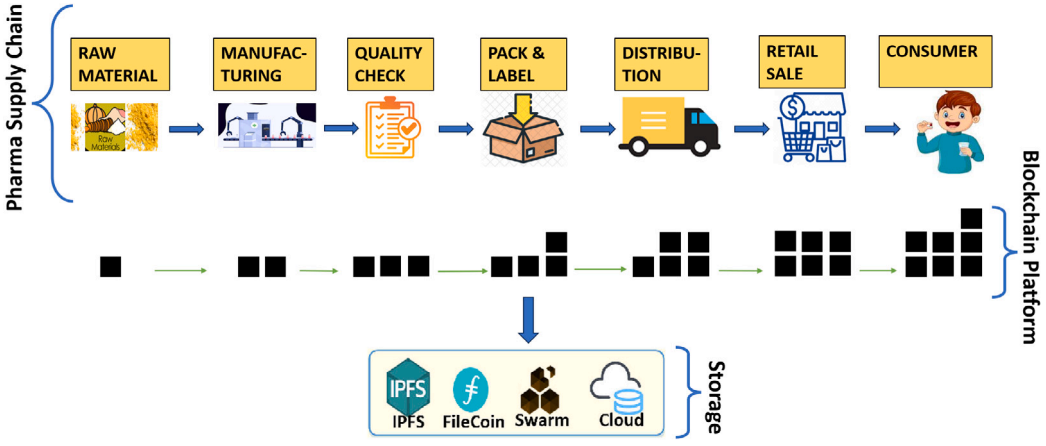


Fig. 4. Architecture of blockchain enabled pharma SCM.

4.4. Constraints

- Delay in data transmission from IoT to Cloud
- Public blockchains used by SCM do not allow programmatic control of blocktime, data size, fee etc.
- Data upload size permissible in the various blockchains are limited
- The consensus algorithms are compute hungry and time consuming
- The time taken for converging to consensus is proportional to the number of blocks involved

5. Proposed solution

In order to address the requirements of real-time processing and control for time critical PSCMS, the four layered multi-tiered architecture is utilized [18]; it comprises of IoT, Mist, Edge, Fog and Cloud computing layers as seen in Fig. 3. The main difference between the layers is in terms of the communication, response, compute, security and storage (CRCSS) capacities. The SCM activities are distributed in the hierarchical architecture, so that the communication delay is reduced and the compute-storage capacity

is exploited to improve the overall response time. This research work uses IoT, four-layered compute-storage infrastructure and Blockchain in the following manner to provide secure, optimized and real-time responsive Pharmaceutical SCM system:

- Use IoT for automation and computing layers for storing and processing data
- Distribute the SCM activities in the hierarchical four layered Mist-Edge-Fog-Cloud architecture, to improve the response time
- Use blockchain to provide secure SCM transactions
- Use blockchain enabled cloud permanent storage for processing or analytics

There have been efforts to improve the overall efficiency of the system by reducing the data transmission delay, and provide lightweight consensus algorithms and smart contracts. While many blockchain-as-a-service (BaaS) are offered on Clouds such as Oracle Blockchain Cloud Service, IBM Blockchain Platform, Corda, Kaledio and Azure Blockchain which leverage Cloud Computing. There are also blockchains that are tuned to Edge computing and Fog Computing. Recent efforts of blockchain integration with fog computing are — FogCoin, IOTA and Aion. Similarly, Hut 8, Solana, and Lumen are examples of blockchain on Edge computing.

Yu et al. [13], proposes a hierarchical edge-cloud blockchain called LayerChain to improve the response and storage of blockchain. Here, the blockchain data is stored in two layers of clouds and edge nodes, thereby reducing the resource requirements and block propagation time, making it well-suited for large-scale low-delay Industrial IoT applications. However, since blockchains are being adapted to edge and fog nodes, the SCM can be made to exploit a hierarchical compute-storage architecture to provide better response time.

5.1. IOT-MIST-EDGE-FOG-CLOUD: IMEFC architecture

A four layered architecture combining the Mist, Edge, Fog and Cloud computing paradigms (explained in Section 2.5) is proposed to run the pharmaceutical supply chain application. This architecture is superior to the two layer LayerChain architecture proposed by Yu et al. in [13]. The LayerChain is a two layer Edge and Cloud computing architecture for improving the response in Industrial IoT. Improving over the LayerChain concept, this work introduces two more intermediate layers of Mist and Fog. By introducing additional Mist and Edge to the two layered architecture helps to better distribute the tasks based on the urgency of response required, storage permanence and compute load. Mist computing offers a cooperative computing paradigm with lateral communication data distribution and compute across nodes at Mist level. However, Edge computing nodes directly connect to the cloud as well as the sensors, but generally do not communicate laterally with other edge nodes and can only work in a single application per setup. When immediate response is required, Mist Computing is most suitable; as the priority is for fast computation, less network delay, and data storage is not mandatory. Mist nodes can be deployed close to the field and perform quick computation and respond to the user; while the data is forwarded to higher layers for storage or more complex processing. The pharmaceutical supply chain shall leverage the IMEFC (IoT-Mist-Edge-Fog-Cloud) architecture to process and analyze data at various levels as follows.

The Pharmaceutical Supply Chain uses the IMEFC architecture at various stages in manufacture plant, packaging, distribution, etc. For example, in the manufacturing plant, it is required to monitor and adjust the temperature and humidity at every ten minutes interval to safeguard the product quality. In such cases, the Mist computing, which is near the data source, can be used for real-time data processing and control. The next stage is the inventory management, product quality assurance and authenticity checks. These activities are more compute intensive and can be performed on Edge Computers. Similarly, route planning, shipment tracking and pharmaceutical logistics can handled in Fog nodes, as it takes inputs from multiple sources which can be converged at the higher level Fog Computing. Finally, the cloud can be used by pharmaceutical businesses for long-term data storage, evaluating historical data, performing advanced analytics, managing large-scale database, data mining and compliance reporting. Hence, different layers of the hierarchical architecture provide appropriate communication-response-compute-secure-storage mechanisms of process in enhancing the performance of the SCM.

5.2. Blockchain based PSCMS

Developing a secure and resilient Pharma SCM necessitates the integration of Blockchain. Real-time pharmaceutical data such as batch numbers, production timestamps, or temperature is collected and input at several phases of the supply chain and should be recorded in the blockchain.

This process is designed as a SCM integrated with blockchain (Fig. 4). The data related to each of the activities in the Supply Chain are stored into blockchain. For example, in the manufacturing process, the quantity, quality, temperature,along with time stamping are uploaded into the supply chain. This becomes an immutable transaction and helps in monitoring and control of the processes and tracking of the goods. Similarly, the distribution process requires to record the package serial number, vehicle number, destination addresses and so on. When there are a large number of components or parameters, the size of the data becomes large and these can be stored in off-chain blockchains with distributed file systems.

5.3. Proposed multi-tier blockchain enabled PSCMS

By combining the Blockchain and multi-layered computational architecture, it is possible to provide security and improve the processing time. The steps for automating the Supply Chain Management by using IoT, Blockchain with off-chain storage in a multi-layered computational architecture, is presented in Algorithm 1.

Algorithm 1 Blockchain for SCM in Multi-tiered Architecture

Input: blockchain_name, transactions_data
Output: storage confirmation
1. Select / Create the Blockchain networks in Edge / Fog / Cloud;
2. Define the authorities which participate in the consensus;
3. Define smart contracts for the SCM;
4. Register the IoT devices and Mist, Edge, and Fog nodes participating in the SCM, to the Blockchain Network;
5. Create a corresponding Blockchain client and connect to the network;
6. Upload SCM process data (on-chain / off-chain mode);
7. return Storage Confirmation

6. Blockchain platforms

The main aspects to improve the performance of the IoT-Blockchain enabled SCM are to improve the response time and secure storage. Hence, we experimentally study few popular blockchain platforms in terms of the upload time, download time, file size, throughput and transaction cost.

6.1. Bitcoin

Bitcoin is a decentralized digital currency that is also known as a cryptocurrency. Unlike traditional government-issued currencies (fiat currencies), Bitcoin runs on a decentralized peer-to-peer network, eliminating the need for intermediaries such as banks. It enables secure and transparent transactions and the blockchain technology that underpins it protects its integrity. The Bitcoin blockchain is a public distributed ledger that records all Bitcoin transactions. It is a series of blocks, with each block containing a set of transactions. These transactions are bundled together and sequentially added to the blockchain. The blockchain acts as a tamper-proof record of each transaction's history, ensuring transparency and security.

Data Storage on Bitcoin Blockchain: While the primary purpose of the Bitcoin blockchain is to record financial transactions, it is also possible to store small amounts of data on it *via* a process known as ''OP_RETURN''. A variety of tasks can be done as part of a transaction using the Bitcoin scripting language. The OP_RETURN opcode is one of these operations. It is a unique technique that lets you attach a short amount of data (up to 80 bytes) to a transaction output. This information is not spendable, which means it does not entail the transfer of Bitcoin between addresses. You would initiate a transaction and include an output using the OP_RETURN opcode to save data on the Bitcoin network. The data to be saved is then passed as an argument to the opcode. This information is often represented using hexadecimal values. Keep in mind that the OP_RETURN opcode provides only 80 bytes of data storage space. This means that only little pieces of data, such as hashes, timestamps, or short messages, can be stored. Further, because every byte of data uploaded to the blockchain requires space, incorporating data in a transaction incurs a cost. Higher data storage entails higher costs. When data is embedded in the Bitcoin blockchain *via* the OP_RETURN opcode, it becomes an indelible part of the blockchain's history. The data is replicated across the whole network of nodes, making changes or removal extremely difficult. This immutability and decentralization are key characteristics of the blockchain technology that Bitcoin pioneered.

The Bitcoin blockchain, however, may not be the most efficient solution for all data storage needs due to restricted space and associated expenses. Other blockchains or decentralized storage solutions, such as IPFS (InterPlanetary File System), may be better suited for big data storage needs. Finally, while Bitcoin is primarily intended for secure peer-to-peer transactions, its blockchain may also be used to store small amounts of data *via* the OP_RETURN opcode. This capacity opens up new options for time-stamping, verification, and proof of existence, but it is critical to understand the constraints and costs involved with Bitcoin blockchain data storage.

Key Parameters: The main aspects of Bitcoin blockchain which affect SCM applications include:

- *Upload Time:* Bitcoin's upload time for files is limited by its block size limits. While the OP RETURN opcode can be used to attach data to Bitcoin transactions, it is normally limited to relatively modest quantities of data, up to 80 bytes. This renders Bitcoin unsuitable for large data uploads or extensive file storage.
- *Download Time:* Downloading specified blocks from Bitcoin blockchain is done by selecting the wallet in the history log from where the blocks should be downloaded. The *getBlock()* method exports the required block to the local client. Download time is proportional to the download speed of internet.
- *Transaction Fees:* Transactions on the Bitcoin network involve the sender paying transaction fees. These fees fluctuate depending on factors such as transaction size, network congestion, and transaction urgency. Miners prioritize transactions with larger fees, resulting in varying transaction costs.

6.2. Ethereum

Unlike Bitcoin's blockchain, which is primarily used to store digital cash, Ethereum's blockchain is intended to allow smart contracts and Decentralized Apps (DApps). Ether (ETH) is the Ethereum network's native cryptocurrency, and it is used to facilitate transactions and pay for computational services on the platform. The Ethereum blockchain is a decentralized, public ledger that records transactions and allows smart contracts to be executed. It goes beyond Bitcoin's capabilities by allowing developers to create and deploy their own blockchain apps, bringing programmability and automation to blockchain technology. The Ethereum blockchain's support for smart contracts is a critical feature. Smart contracts are self-executing agreements in which the contract terms are directly encoded into code. When established criteria are met, they automatically execute and enforce the contract's provisions. This enables the trustless and tamper-resistant automation of a wide range of operations, including financial transactions, supply chain management, and others.

Data Storage on Ethereum Blockchain: A data storage component is included in Ethereum smart contracts. They can directly store data variables and structures on the blockchain. Depending on the contract's design, these variables can be either public or private. When data is saved in an Ethereum smart contract, it becomes immutable and resistant to tampering. This is a key feature of blockchain technology. The data contained in a transaction becomes a permanent part of the blockchain's history once it is confirmed and added to a block. Gas fees must be paid for storing data on the Ethereum network. The processing resources required to process and record the transaction on the blockchain are covered by the gas costs. The higher the gas expenses, the more complex the data storage procedure (for example, writing big volumes of data). As a result of its support for smart contracts and decentralized applications, Ethereum's blockchain offers a more adaptable setting for data storage. While data storage has gas costs and scalability issues, Ethereum presents a powerful foundation for developing decentralized and transparent solutions that can transform numerous industries.

Key Parameters: The main parameters of Ethereum with respect to SCM applications are:

- *Upload Time:* Uploading files to Ethereum can be achieved through smart contracts, which offer greater flexibility compared to Bitcoin. However, this flexibility comes at a cost. Large file uploads can result in significant gas costs, making Ethereum less cost-effective for extensive file storage. Developers must carefully manage gas usage to avoid high expenses.
- *Download Time:* Ethereum has two modes of storage - on-chain and off-chain. The on-chain download time will be 2–3 s as it involves only the metadata. For Off-chain mode, the time mainly depends on size of file being downloaded from the IPFS; the metadata download time from blockchain takes only 2–3 s. Generally the download time is slightly more than upload time to IPFS.
- *Transaction Fees:* In Ethereum Transaction Fee is called the Gas Price. Gas prices on the Ethereum network determine the speed and cost of transaction processing. Users must specify the gas price they are willing to pay to prioritize their transactions. High gas prices can increase the cost of file uploads, making Ethereum potentially expensive for data storage.

6.3. Filecoin

Protocol Labs introduced Filecoin, a decentralized storage network and cryptocurrency, in 2020. It intends to use blockchain technology to build a global marketplace for data storage and retrieval. In contrast to typical centralized storage systems, Filecoin employs a distributed model in which individuals and companies can rent out excess storage space in exchange for the native cryptocurrency, FIL. This results in the formation of a decentralized network of storage providers and customers, providing safe, dependable, and cost-effective data storage. The Filecoin network is based on a blockchain, which acts as the basis for its storage marketplace. This blockchain tracks storage transactions, data recovery, and storage provider incentives. It incorporates a number of novel proof procedures to ensure that storage providers keep their promises and retain data accurately.

The decentralized storage model of Filecoin has various benefits, including Data being spread across many storage providers, and lowering the chance of data loss due to hardware failures or other problems. The network employs cryptographic algorithms to secure the integrity of stored data, ensuring that it is not altered. Users can find storage providers who meet their requirements, resulting in a competitive market for storage services.

Data Storage on Filecoin Blockchain: Clients are network users who want to store data on the network. They pay storage providers in FIL tokens in exchange for storage services. Storage providers or miners are individuals or companies who supply storage capacity. They use FIL tokens as collateral and get rewards for storing data successfully. Clients and storage providers form storage agreements, defining terms such as data volume, duration, and pricing. Cryptographic proofs are used by storage providers to indicate that they are storing the data as claimed. The *Proof of Replication* ensures that data is truly reproduced, whereas the *Proof of Space-Time* provides ongoing storage across time. Clients can obtain their data by submitting retrieval requests and paying the necessary fees. Miners respond to these queries by delivering the data requested.

The decentralized storage capabilities of Filecoin pave the path for a variety of data storage use cases such as supply chain and logistics. Thus, Filecoin is a decentralized storage network that employs blockchain technology to provide a global marketplace for data storage and retrieval. Filecoin intends to revolutionize data storage by combining a distributed approach, cryptographic proofs, and an incentivization model, providing security, redundancy, and efficiency for a wide range of users and applications.

Table 6
Detailed comparison between Bitcoin, Ethereum and Filecoin.

Aspects	Bitcoin	Ethereum	Filecoin
Primary function	Digital currency	Smart contract, Blockchain	Decentralized storage
Data storage	Limited, meta/small data	Limited, Smart-contract based	Decentralized, Scalable
Transaction cost	Can be expensive	Higher gas fees	Varies based on market, filesize
Scalability	Limited	Scaling effort (ETH 2.0)	Scalable for data volume
Data integrity	Limited	Blockchain-based, Immutable	Cryptographic proof for storage
Economic incentives	N/A	N/A	Rewards storage miners
Smart contract	N/A	Business logic, Automation	N/A
Data privacy	Limited	Private transaction	Varies, Focus on storage
Use cases	Currency	Automation, Business logic	Data storage/retrieval
Applications	Finance	Finance, Gaming, SCM	Data storage
Suitability for SCM	Limited for extensive data	Automation and Data storage	Suitable due to scalability
On/Off-chain	On-chain	On-chain	Hybrid, On and Off-chain
Data size limit	Transactions with limited data attachment	Gas constraints apply to smart contracts	Designed for more extensive file storage
Decentralization	Decentralized, but mining power can be concentrated	Moving towards Ethereum 2.0 for enhanced decentralization	Decentralized storage and retrieval are prioritized
Censorship resistance	Transactions that are not subject to censorship	Smart contracts and transactions can be resistive	Data storage intended to withstand censorship
Developer tools	Scripting language and tools are limited	Smart contract developer tools and frameworks that are robust	Interaction tools for storage miners and the network
Adoption and Ecosystem	Strong popularity as a digital money, but restricted in other applications	A varied ecosystem with numerous use cases, including supply chain	Concentrated on data storage, decentralized web popularity is increasing
Interoperability	Limited interoperability, mostly a cryptocurrency	Smart contracts enable interoperability.	Emphasis is on data storage, with the possibility of integration with other platforms.
File retrieval	Simple metadata retrieval	Smart contract retrieval can be complicated	Created for quick file retrieval
Transaction finality	High degree of transaction finality	Finality is contingent on block confirmation	The consensus procedure determines the outcome.
Network upgrades	Due to consensus, network improvements might be time-consuming	Upgrades are ongoing, and the move to Ethereum 2.0 is underway	Possibility of protocol enhancements
Governance and Community	Decentralized governance	Transitioning to Ethereum 2.0, governance models evolve	Miners and stakeholders are involved in governance
Upload/Download time	Limited due to block size restriction	Large files incur significant gas costs	Variable based on miners and demand
Transaction fees	Costly	Gas prices influence both speed and cost	Storage miners set pricing

Key Parameters: The parameters of Filecoin scalable data storage are:

- *Upload Time:* The upload time of Filecoin varies and is determined by a number of factors, including the selection of storage miners, network demand, and the size of the data being uploaded. Filecoin's marketplace approach allows customers to select miners based on their storage requirements and budget.
- *Download Time:* Filecoin works in off-chain mode with IPFS. The download time depends on size of file being downloaded. Generally the download time is slightly more than upload time to IPFS.
- *Transaction Fees:* Filecoin, unlike Bitcoin and Ethereum, does not use a gas-based fee scheme. It instead follows a marketplace approach in which storage miners set pricing for their services based on supply and demand dynamics. This strategy enables consumers to choose storage options that fit their budget and needs.

Hence, the selection of a blockchain platform for file uploads in supply chain management is influenced by a number of criteria, including the project's specific requirements as depicted in Table 6. Bitcoin has restrictions on data size and the network concentration of mining power. Ethereum sets gas restrictions on smart contracts, which might limit the amount of data that can be processed within a transaction. Ethereum is planning to convert to Ethereum 2.0, which aims for enhanced decentralization to address some of the concerns about centralization. Filecoin, which is particularly intended for greater file storage, takes a different approach. It emphasizes decentralized storage and retrieval, making it suited for data storage, and it is designed to efficiently resist censorship.

In terms of developer tools, Bitcoin has a limited scripting language and tools, whereas Ethereum has a solid ecosystem with substantial developer tools and libraries designed expressly for smart contracts. Filecoin provides tools for dealing with storage miners and the network, allowing for easier data storage and retrieval. In terms of ecology and adoption, Bitcoin is widely used as a digital currency but has few applications beyond that. In contrast, Ethereum has a robust ecosystem with a wide range of applications, including supply chain management and decentralized finance (DeFi). While Filecoin is primarily used for data storage, it is gaining popularity in the context of the decentralized web. The interoperability of these platforms varies. Bitcoin is primarily a cryptocurrency with limited compatibility with other blockchain networks. With its smart contract features, Ethereum promotes interoperability by allowing interactions with other blockchain platforms. Filecoin primarily focuses on data storage, although it may eventually integrate with other blockchain systems. When it comes to file retrieval, Bitcoin provides simple metadata recovery, whereas Ethereum's retrieval procedure can be complicated, frequently including smart contracts. Filecoin, on the other hand, is expressly intended for efficient file retrieval, making it appropriate for decentralized storage and data retrieval use cases. The finality of transactions varies across these networks. Bitcoin provides a high level of transaction finality, giving users a sense of confidence. The finality of Ethereum is dependent on block confirmation, which might vary in terms of timing and confidence. The finality of Filecoin, like Ethereum, is determined by the consensus mechanism and network dynamics.

Each platform faces its own set of obstacles when it comes to network improvements. Because of the need for consensus among miners and stakeholders, Bitcoin network updates can be slow. Ethereum is undergoing continuous improvements, with a transition to Ethereum 2.0 occurring to address scalability and sustainability problems. Filecoin's protocol has the ability to develop as it evolves. Finally, the governing arrangements of various platforms differ. Bitcoin is based on decentralized governance, which allows users and miners to vote on protocol updates. Ethereum is adopting governance structures to achieve greater decentralization as it transitions to Ethereum 2.0. The governance of Filecoin includes stakeholders and miners, ensuring that crucial decisions are decided cooperatively within the ecosystem.

Ultimately, the selection of a blockchain platform for file uploads in supply chain management is influenced by a number of criteria, including the project's specific requirements, the extent of data to be kept, the desired level of decentralization, and the available budget. While Bitcoin and Ethereum have established blockchain infrastructures, they have file storage restrictions and can be influenced by changing fees and delays. Filecoin, on the other hand, is designed specifically for scalable and efficient data storage, making it an attractive option for supply chain applications that require significant and secure file uploads.

7. Implementation

7.1. Ethereum

A potent method to improve transparency and traceability inside the supply chain ecosystem is to upload supply chain data to the Ethereum blockchain. This procedure entails developing and deploying smart contracts on the Ethereum blockchain, which store and handle supply chain data. Uploading supply chain data to the Ethereum blockchain is a complicated procedure that comprises several important steps. By effectively carrying out this process, we can improve the transparency, security, and traceability of the supply chain management system. The various steps to establish the Ethereum environment are given here under.

(i) *Establishing Development Environment*: It is critical to fully prepare the development environment before commencing the adventure of uploading supply chain data to the Ethereum blockchain. Truffle, Ganache, and Infura are three crucial software tools for this aim.

- *Truffle:* Truffle is a complete Ethereum development framework. It offers a set of tools to help with smart contract creation, deployment, and management. To get started, use the npm (Node Package Manager) command npm install -g truffle to install Truffle.
- *Ganache:* Ganache is a local blockchain emulator that provides a controlled environment for testing smart contracts. It includes Ethereum accounts with synthetic Ether for testing purposes.
- *Infura:* Infura is a critical service that provides access to Ethereum nodes without requiring us to run our own Ethereum node. To begin, you must first register an Infura account and obtain an API key. This API key is required for interactions with the Ethereum network.

(ii) *Creating and Compiling Smart Contract*: After setting up the development environment, we can begin working on the smart contract. A smart contract developed in Solidity (the Ethereum programming language) describes the logic required to upload supply chain data to the Ethereum network. The smart contract contain the rules and procedures that govern the storage and retrieval of supply chain data. This stage entails carefully considering the individual supply chain requirements and developing a contract that handles them efficiently.

(iii) *Smart Contract Deployment*: The final step is to deploy the smart contract on the Ethereum blockchain after it has been written and extensively tested. If it is in the development or testing phase, Ganache is likely to be used to publish it to a local blockchain. Deployment for production use is either on the Ethereum mainnet or a testnet. In either instance, we need to connect to the Ethereum network and deploy the smart contract using Infura. It is critical to ensure that the contract is fully financed with Ether to cover deployment transaction expenses.

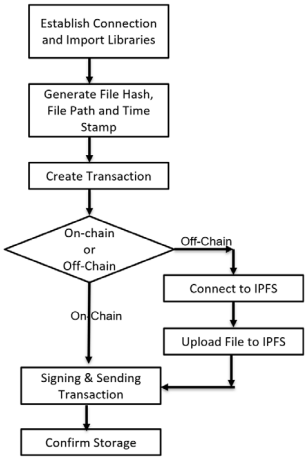


Fig. 5. On-chain and off-chain storage in Ethereum.

(iv) SHA-256 hashing of Data: Data integrity and security are two of the most important components of uploading supply chain data to the Ethereum blockchain. This is where the SHA-256 algorithm comes into play for data hashing. The process of turning data into a fixed-length hexadecimal code (typically 64 characters) that serves as a unique representation of that data is known as hashing. It is critical to guarantee that the data remains unmodified and private throughout its blockchain journey. Any supply chain data should be hashed with SHA-256 before being uploaded to the blockchain. This step is critical for protecting the data against manipulation and illegal access. Before the data is delivered to the smart contract for storage, it is normally hashed in the application code.

(v) Interacting with Smart Contract: A contract can be initiated to store the supply chain data on the Ethereum blockchain after the data is hashed. Typically, a smart contract function is specifically created for this purpose. An Ethereum client library (e.g., web3.js for JavaScript) is used within the application code to carry out this interaction. This interaction entails sending a transaction to the smart contract, which then performs the chosen function while passing the hashed supply chain data as a parameter. After receiving the transaction, the smart contract will execute the function and store the data on the blockchain.

The supply chain data can be successfully uploaded to the Ethereum blockchain by following these well-defined processes and using the necessary software tools such as Truffle, Ganache, and Infura, as well as rigorous data hashing using SHA-256. This method improves openness, security, and trust in the supply chain management system. In addition, it also encourages accountability and efficiency, which benefits the firm and all stakeholders.

7.1.1. Storing on Ethereum blockchain - On-chain or off-chain

Data storage is just one of the many uses for the decentralized, unchangeable ledger that blockchain technology offers. Using the web3 library and the Infura API, the procedure to hash a file's contents, store the hash and a timestamp on the Ethereum blockchain, and communicate with the Ethereum network is explained in [67]. Off-chain transactions are any non-transactional data that is too large to be stored in the blockchain efficiently, or, requires the ability to be changed or deleted [68]. The transactions may occur outside of the blockchain, entail lower fees, immediate settlement, and greater anonymity than on-chain transactions. The flowchart in Fig. 5 provides a pictorial representation of the various steps involved in storing data in Ethereum in On-Chain or Off-Chain modes.

Establishing a connection and importing libraries: The required libraries such as time for timestamp retrieval, hashlib for hashing, and web3 are imported for the Ethereum interface. Both the Ethereum address and the endpoint URL for the Infura API are specified. Through the Infura service, a connection to the Ethereum network is made, and the connection's status is verified.

Calculating File Hash: The SHA-256 hash of the data in a given file is calculated by the calculate_file_hash function. To minimize memory problems with huge files, it reads the file in digestible pieces (4096 bytes), updating the hash on the go, and the hexadecimal formatted hash values are returned.

File Path and Hash Calculation: The file_path variable must contain the real file path. The contents of the file are then computed using the calculate_file_hash function. The integrity of the file is confirmed using this hash.

Generating Timestamp: The *time.time()* function creates a Unix timestamp. This timestamp shows the seconds that have transpired since the Unix epoch, or January 1, 1970. It serves as a date stamp on the file's blockchain entry.

Creating a Transaction: The Ethereum blockchain transaction is ready to contain the file hash and timestamp. The transaction dictionary comprises the following crucial data:

- *to*: The Ethereum address of the recipient (to whom you want to send the transaction).

- *data*: The data you want to include in the transaction, in short. Here, web3.toHex() is used to convert the string that results from concatenating the file hash and timestamp into hexadecimal representation.
- *gas*: The most gas that can be used for the transaction. Ethereum uses a processing unit called *gas*, and transactions need gas to be carried out.
- *gasPrice*: The cost for one unit of blockchain gasoline.
- *nonce*: A unique number connected to the user account that prevents repeat transaction attacks.

Connecting IPFS: Using the ipfshttpclient library, the script connects to the IPFS network. IPFS (InterPlanetary File System) is a decentralized protocol that provides a content-addressable system for storing and distributing files.

Uploading File to IPFS: Using the *ipfs_client.add* method, the calculated hash is then utilized to upload the file to the IPFS network. This method uploads the file to the IPFS network and returns a unique hash of the uploaded data. This hash can be used at any time to get the file's content from the IPFS network.

Signing and Sending Transaction: We can only send transactions from our Ethereum address since the transaction is signed using your private key. The *sendRawTransaction* method is then used to send the signed transaction to the Ethereum network. A transaction's unique identifier, and the resulting transaction hash, are printed on the console.

7.2. Filecoin

Usage of Filecoin blockchain for supply chain is a meticulously planned procedure that leverages the potential of decentralized storage to improve data security, accessibility and lifespan.

(i) Prepare the Environment: Establish the development environment before saving supply chain data on the Filecoin blockchain. We require *Node.js 14* or higher, as well as *NPM 7* or higher. These are required before running the scripts and interacting with the Filecoin network programmatically.

(ii) Create a Filecoin Account: A Filecoin account is required to use the Filecoin network for decentralized storage. This account gives the credentials and access that is required to store and retrieve data from the network. Typically, a Filecoin account can be created using the Filecoin network's official website, where one can sign up and acquire the account details.

(iii) Generating API Token: Filecoin, like *web3.storage*, frequently requires the usage of API tokens for authentication and authorization. These tokens are required for uploading and managing data and allows to interact with the Filecoin network programmatically. Generate an API token by going to the Filecoin account settings and following the prompts to create a new token. Keep in mind that the API token allows access to the user's account. Hence, the API Token should be protected.

(iv) Creating the Upload Script: With the environment in place and the API token in hand, now develop an upload script. This script interfaces with the Filecoin network, starts the upload process, and obtains a Content Identifier (CID) for the saved data. The upload script is often written in JavaScript and interacts with the network *via* the Filecoin client libraries or SDKs. This script takes the API token, the path to the file or data to be uploaded, and any additional parameters we provide. The data is then sent to the Filecoin network to begin the upload process.

(v) Initializing the Upload: When the upload script runs with the correct parameters, the data is uploaded to the Filecoin blockchain(IPFS Storage). The script uses the API token to authenticate with the Filecoin account and locate the data to be stored using the path provided. Once the upload is complete, the script returns a CID, which is a unique identifier for the Filecoin network data.

(vi) Obtaining Stored Data: The simplicity of retrieval is one of the benefits of adopting Filecoin for supply chain data storage. To access the saved data, simply enter the CID that was provided during the upload procedure. Data can be retrieved from the Filecoin network (IPFS Storage) by creating the necessary URL or request, making it available for the supply chain management system or any other applications that require access.

(vii) Data Protection and Privacy: While Filecoin offers extensive storage possibilities, it is critical to consider data security and privacy. To protect sensitive information, make sure the data submitted to Filecoin is properly encrypted and secured. Further, use best practices for access control to limit retrieval of the stored data.

(viii) Monitoring and Upkeep: Continuous monitoring and maintenance are essential for any data storage solution. Check the status of the saved data on a regular basis, including its availability and integrity. Filecoin provides tools and APIs for maintaining and monitoring the data, ensuring that it is always available and dependable.

To sum up, adding supply chain data to the Filecoin blockchain and storing it there improves data security and accessibility. The benefits of decentralized storage can be utilized while properly managing the supply chain data by following these steps, from setting up the infrastructure to designing upload scripts and maintaining data security. With its powerful storage capacity and ease of usage, the Filecoin network is a great resource for enterprises looking for safe and dependable data storage solutions.

7.2.1. Storing on Filecoin

Filecoin is built on top of IPFS and provides off-chain method of storage. The Filecoin ecosystem provides clients such as lotus to provide commands and APIs for storing and retrieving the files. The steps involved in storing data to Filecoin are given below.

(i) Register to the Filecoin storage: Register with the IPFS storage server, say *chainsafe*, and set up a wallet. Once a wallet is created with some Filecoin tokens, the client can be used for working with Filecoin.

(ii) *Create a Filecoin client and connect to the network*: Methods like *new.filecoin* and *filecoin.connect* are used to connect to the Filecoin network.
(iii) *Upload your data*: Upload the data to the Filecoin network using the upload method. The file size, duration of storage and transaction fees should be specified.
(iv) *Storage Confirmation*: The Filecoin returns the IPFS hash after storing the file. The download method can be used for accessing the stored file
(v) *Download the data*: Since reading the data stored on blockchain may be required for forensics or analytics, this can be an optional step. A blockchain explorer is a utility in the client system to access the data stored blockchain. For example, the explorer for Ethereum is Etherscan, FileCoin is FilFox, Solana is Solscan and that for Bitcoin it is Blockchain.com's Bitcoin Explorer. It supports APIs such as the Get method to programmatically access blockchain data.

8. Experimentation and performance analysis

8.1. Platform

The experiment was conducted for three public Blockchain platforms — Bitcoin, Ethereum and Filecoin by connecting from a Intel Core i5 PC with 16 GB RAM, 1TB Hard disk connected to Internet link with speed of 1 Gbps. The blockchain clients are downloaded and run on the PC. The blockchain wallet is obtained with tokens. The client connects to other nodes of the blockchain network. The main aim is to study the feasibility of the blockchains for suitability to complex SCM applications. The blockchains are selected from the perspective of their capacity to handle data size. Small, medium and large data sets in the range of 100 Bytes to 1.5 Giga Bytes are used for uploading and downloading to the blockchains over the internet link of 1 Gbps. For Ethereum both on-chain and off-chain methods are experimented; for Filecoin it is an off-chain mode and relies on a private IPFS.

8.2. Parameters

Since the SCM applications may have different sizes of data for each of the processes (such as sensor data, raw material details, packaging information, payment details, etc.) it is required to study the blockchains behavior for different data sizes. Small data in Bytes, medium data in KBs and large data in the range of MBs are used for uploading and downloading to the blockchains. The main parameters tested are upload/download time and transaction costs.
(i) Upload Time: The Upload time is generally the time taken for sending the file from a device to the server. In the context of Blockchain, the upload time is dependent on the data size, network bandwidth and involves storing the data inside each block on the chain (blocktime). In other words, upload time is also influenced by the type of consensus algorithm being executed before the block gets committed.
(ii) Download Time: The Download Time is the time taken for the block to be downloaded. It is dependent on the network download speed. Typically, the download of data from the chain of blocks is done in units of blocks. The *getblocks(locator(chainHead), 0)* function uses the *blockLocator()* function checks the headers of all blocks and locates the block in the blockchain and then returns the block. The Downloader sends *getdata()* request and the Responder sends the block of data. For large data, which uses off-chain methods, it involves downloading metadata from the blockchain and the file from IPFS. Since metadata download is only few seconds, the download time represents the time to get unsealed data from the IPFS to the client system.
(iii) Transaction Fees: Transaction fees is the cost paid for a transaction on the public blockchain networks. Bitcoin charges in terms of BTC (Bitcoin Transaction Fees) and Ethereum uses ETH units. The fees are governed by tariff slabs of the blockchain platforms. The total cost is the fee rate multiplied with the size of the transaction.

8.3. Performance analysis

Figs. 6–8 show the upload and download time for small medium and large data on the three blockchains in on-chain and off-chain mode. The experiment examines which blockchains supports larger data and the amount of upload/download time they take. In Bitcoin the data is stored on-chain, while Ethereum works in both on-chain and off-chain modes; and in Filecoin data is stored off-chain. Different sized files from few bytes to few GigaBytes are taken for studying the upload/download time, keeping the network bandwidth constant. The impact on upload time becomes significant with the increase in file size.

For *small data* all the three blockchains — Bitcoin, Ethereum and Filecoin are feasible. For *medium data* (hundreds of KBs), Ethereum on-chain, Ethereum off-chain and Filecoin are feasible in terms of transaction cost and time. For *large data* (GBs) Ethereum off-chain and Filecoin are possible. Bitcoin has a higher blocktime and hence uploading even small data takes nearly 600 s on Bitcoin, while Ethereum upload time is about 150 s and the Filecoin is about 120 s. For medium sized data upload, Ethereum on-chain as well as Ethereum off-chain modes take between 180 to 240 s. Filecoin takes about 200 s for uploading medium size data, which is lesser time compared to Ethereum. For large data upload, it is seen that Filecoin performs better than Ethereum off-chain. The Filecoin platform took 380 s to upload 1.4 GB data file, while Ethereum off-chain takes nearly 420 s. The graphs show that only small file sizes are supported on Bitcoin, while medium sized files are supported on Ethereum (on-chain/off-chain) and Filecoin, and large files work well in off-chain mechanisms.

The experiment is extended to download the same data that is uploaded. The download time for small file sizes is about 300 s for Bitcoin and 100 s for Ethereum on-chain. The download time for small file sizes for off-chain Ethereum and Filecoin are about 90 s.

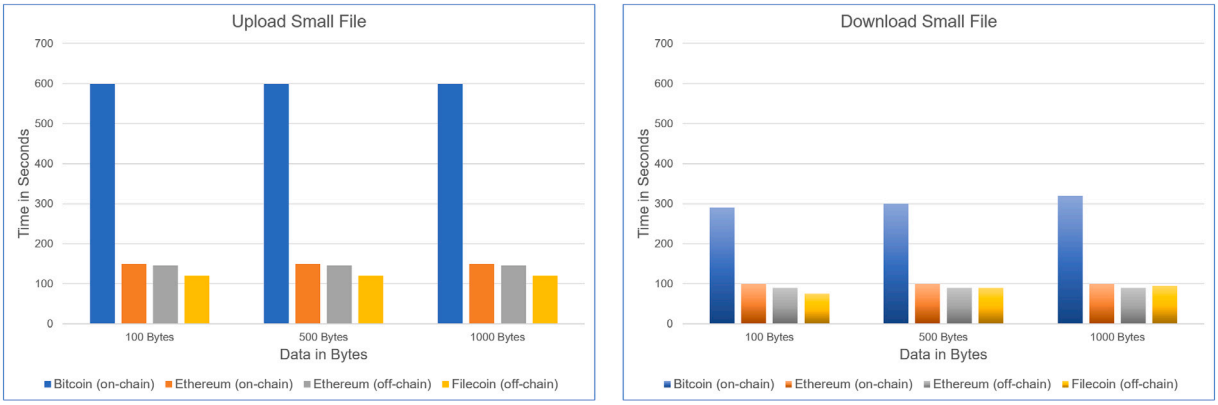


Fig. 6. Upload and download time for small file size in Bitcoin, Ethereum (on- off-chain), Filecoin.

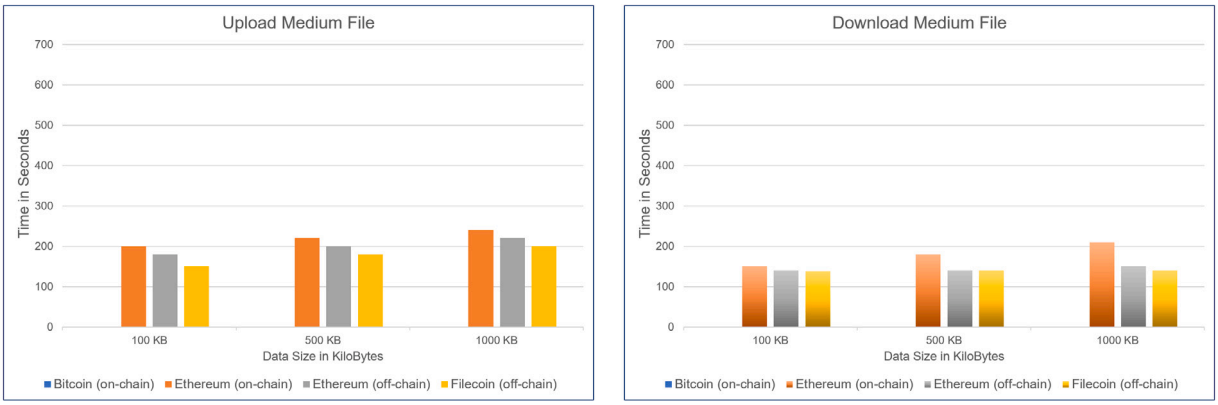


Fig. 7. Upload and download time for medium file size in Ethereum (on- off-chain), Filecoin.

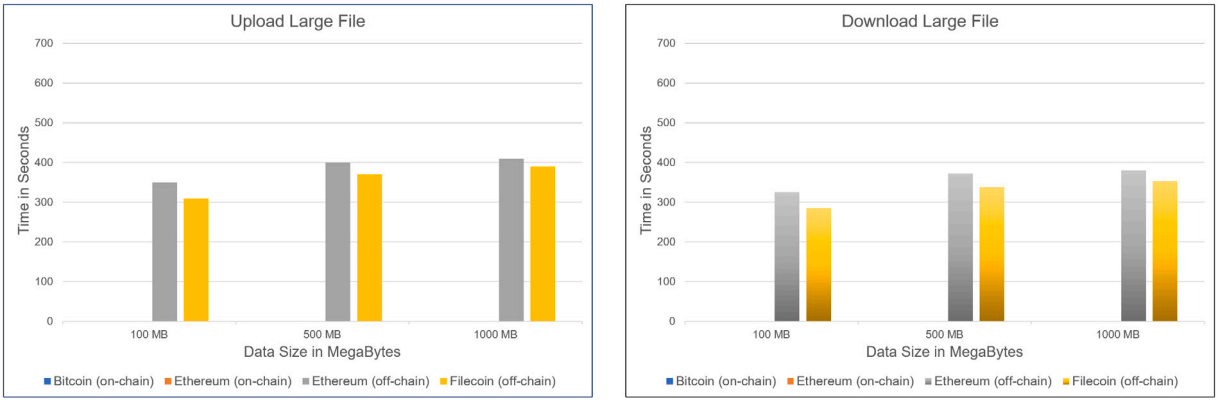


Fig. 8. Upload and download time for large file size in Ethereum (off-chain), Filecoin.

For medium data, the download time is around 150 to 200 s in Ethereum on-chain, Ethereum off-chain and Filecoin. In case of larger file sizes, the data are available in off-chain mode blockchain platforms of Ethereum off-chain and Filecoin. The download of 1.4 GB unsealed data from Filecoin IPFS took 560 s; and 590 s for Ethereum off-chain. It is observed that the download time for off-chain mode blockchains is about 7% faster compared to the corresponding upload. Since download speed in any network is faster, and as there is no consensus processing, ostensibly the data is retrievable (downloadable) in few seconds. But in the blockchain platforms, the entire block should be downloaded using the block locator and get data functions. In off-chain mechanisms, the download involves downloading metadata from the blockchain and the file from IPFS. This explains the performance of download operation.

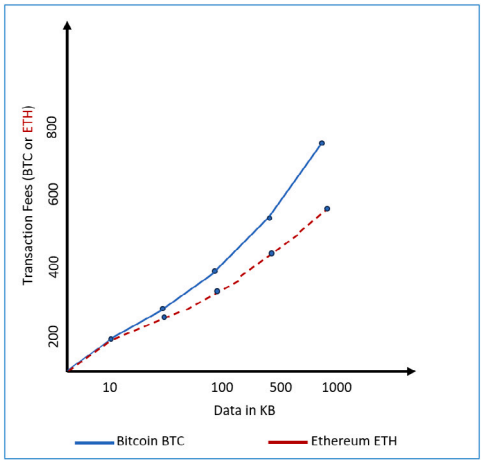


Fig. 9. Bitcoin and Ethereum transaction cost.

Transaction fees is the cost you pay for a transaction on the public blockchain networks. Bitcoin charges are in terms of BTC (Bitcoin Transaction Fees) and Ethereum uses ETH units. Fig. 9 depicts the transaction fees in Bitcoin (BTC) and Ethereum (ETH), charged for small file size upload as it is quite infeasible to pay for large file size uploads for experimentation. Bitcoin is slightly costlier compared to Ethereum. The Filecoin was run in off-chain mode and did not use gas fees scheme. The transaction cost increases with increase in data size.

It is observed that Filecoin accommodates large data files, while Ethereum supports medium sized files, and Bitcoin does not support file uploads (it uploads only transaction data). As the Pharmaceutical SCM system requires to store an aggregation of several relevant IoT data in the Cloud for analytics, the Filecoin platform is the obvious choice, capable of handling big sized data files. It is inferred that when larger file sizes are involved, as may be in the case for pharmaceutical SCM systems, Filecoin is a more suitable solution compared to Bitcoin or Ethereum. The blockchain platform are selected from the perspective of their capacity to handle file size, i.e., Bitcoin is considered for small sized transaction data, while Ethereum for small and medium sized files, and Filecoin is considered for large data files storage marketplace. Data of reasonable size, in the range of 1 GB, is taken for upload and download. The Ethereum Off-chain mode and Filecoin offer reasonably good upload/download time, making them potential candidates for SCM applications. The upload and download time increases proportional to the increase in data size. Public blockchains are scalable for many transactions as they are hosted on public clouds. Since FileCoin offers off-chain mode without gas fees, it is suitable for the SCM application.

8.4. Discussions

The Pharmaceutical Supply Chain Management (SCM) is designed by integrating IoT, blockchain and cloud computing. The design consists of a two-pronged approach: (i) distribution of SCM tasks in the hierarchical IMEFC architecture and (ii) blockchain enabled SCM process management and data storage. This IMEFC architecture addresses the real-time processing needs by distributing SCM activities across Mist, Edge, Fog, and Cloud layers, optimizing communication, response, compute, security, and storage capacities. The IMEFC architecture facilitates distribution of processing and storage to improve the response of time-critical applications [13,18]. The response time for a job is the summation of the data transfer time to the Mist/Edge/Fog/Cloud layer and the processing/storage time in the respective layers. Following are few aspects that need to be deliberated and addressed for real-world implementation:

Complexity and Implementation Cost: The initial establishment cost of the IMEFC architecture will be paid up by the benefits of real-time response for multiple real-world applications. Other benefits of IMEFC architecture are dynamic spatio-temporal processing, region-wise processing and reducing the burden on cloud by preprocessing/filtering. Applications such as automatic vehicles which require dynamic spatio-temporal calculations use Mist nodes for interaction between devices. The Fog computing layer is useful for performing region-wise processing for data from specific subgroups of IoT devices.

Scalability of Blockchain: Blockchain uses individual nodes to process each transaction on the blockchain network. Plasma and Sharding Plasma are two popular solutions that eliminate the need for every node to process each transaction on the network. Other efforts to address scalability of Blockchain are by using Fog (FogCoin, IOTA, Aion) and Edge (Hut 8, Solana, Lumen) computing.

Interoperability Concerns: Typically, every blockchain works in its own way and interoperability among different Blockchain-SCM systems is an open issue that needs to be researched.

Data Privacy and Security: Homomorphic Encryption and Privacy Preserving Encryption techniques are used to protect individual privacy while transferring/storing data in the SCM. The Differential Privacy technique adds noise to the data collection to ensures

that individual privacy is protected, while allowing the data collection to be used for statistical analysis. Further study is required to understand the impact of laws such as General Data Protection Regulation (GDPR), which give people the right to decide how their data is stored or deleted.

Compliance with Regulatory Requirements: Pharmaceutical practitioners and stakeholders of healthcare sector should be consulted to understand the challenges related to regulatory requirements, privacy protection and sensitive information disclosure on public blockchains.

Resource Constraints: The computational capacity, storage capacity and permanence of storage increases from Mist to Cloud. Balancing computational workload and data storage across the hierarchical architecture while optimizing resource utilization is challenging. The tasks are first distributed according to the urgency of response and the semi-processed data are sent to higher layers for offline tasks and permanent storage. Hence, the hierarchical architecture is suitably exploited to provide superior Compute-Response-Communication-Security-Storage (CRCSS).

Adoption and Change Management: Though automated SCM produces improved product quality, quantity and at reduced time, there are challenges for the Blockchain-based SCM to be accepted by the user community. The various stakeholders are hesitant to trust the IoT-Cloud-Blockchain SCM. Data security, privacy, governance policy and interoperability are some key aspects hindering its adoption.

Maintenance and Upkeep: Daily monitoring and periodic auditing of the operational system is a must for such real-world critical systems. Hardware updates, software patches and the preventive maintenance schedule are part of the maintenance plan.

Addressing these concerns requires a holistic approach, involving collaboration between technology providers, regulatory bodies, industry stakeholders, and academic institutions. It is essential to conduct thorough feasibility studies, pilot projects, and continuous evaluation to identify and mitigate potential challenges before deploying IoT-Cloud-Blockchain enabled Pharmaceutical SCM systems in real-world settings.

9. Conclusion

Supply Chain Management systems are utilized in almost every domain such as agriculture, textiles, chemical industry, construction, processed food, healthcare, aerospace and defense, for professional monitoring and management of business processes to improve productivity and profit margin. The important requirements for an automated SCM system are — security, integrity, availability, traceability, scalability and real-time response. The integrated IoT-Cloud-Blockchain platform is a de facto choice for secure SCM systems as it ensures time-stamping, authentication, processes coordination, non-repudiation, aiding commercial transactions, authentication, data integrity, and security for transactions storage. A Pharma SCM based on the IMEFC architecture has been designed to distribute SCM tasks processing to Mist, Edge, Fog and Cloud and provide real-time response for SCM applications. The blockchain technology is used for transactional and data security; the Cloud platform supports execution of distributed consensus algorithm and data storage. The proposed system enhances communication-response-compute-security-storage (CRCSS) performance of the critical SCM tasks. In a Pharmaceutical SCM, the amount of data generated is considerably large as it involves several operations and multiple parameters. Hence, this research focused on determining a scalable, fast and cost-effective Blockchain suitable for the Pharmaceutical SCM. Experimental analysis revealed that Filecoin is faster and cost-effective and accommodates larger file sizes compared to Ethereum and Bitcoin. However, given the sensitive nature of the Pharma industry, privacy concerns, drug recall issues, and regulatory requirements, it is important to get this system validated by all the stakeholders before deploying as a productional system.

A study of the scalability, interoperability, maintenance and regulatory compliance needs to be addressed in future. An analytics application for Pharmaceutical sector needs to be developed using Deep Learning techniques.

CRediT authorship contribution statement

Mangala N.: Writing – review & editing, Writing – original draft, Software, Resources, Project administration, Methodology, Investigation, Conceptualization. **Naveen D.R.:** Software. **B. Eswara Reddy:** Supervision. **Rajkumar Buyya:** Writing – review & editing. **Venugopal K.R.:** Supervision, Conceptualization. **S.S. Iyengar:** Supervision. **L.M. Patnaik:** Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] L.S. Vailshery, Number of IoT connected devices worldwide from 2019 to 2023, with forecasts from 2022 to 2030, 2023, https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide.
[2] S. Saha, IoT in supply chain market outlook, future market insights, 2023, https://www.futuremarketinsights.com/reports/iot-in-supply-chain-market.
[3] S. Ahmed Khan, S. Kusi-Sarpong, H. Gupta, F. Kow Arhin, J. Nguseer Lawal, S. Mehmood Hassan, Critical factors of digital supply chains for organizational performance improvement, IEEE Trans. Eng. Manage. (2021) 1–15.
[4] H. Malik, T. Anees, M. Faheem, M.U. Chaudhry, A. Ali, M.N. Asghar, Blockchain and internet of things in smart cities and drug supply management: Open issues, opportunities, and future directions, Internet Things (2023) 100860–100874.
[5] A. Ghadge, M. Bourlakis, S. Kamble, S. Seuring, Blockchain implementation in pharmaceutical supply chains: A review and conceptual framework, Int. J. Prod. Res. 61 (19) (2023) 6633–6651.
[6] C. Yang, S. Lan, Z. Zhao, M. Zhang, W. Wu, G.Q. Huang, Edge-cloud blockchain and IoE-enabled quality management platform for perishable supply chain logistics, IEEE Internet Things J. 10 (4) (2023) 3264–3275.
[7] Reuters, Boeing urges airlines to inspect 737 MAX planes for possible loose bolt, 2023.
[8] K. Abbas, M. Afaq, T. Ahmed Khan, W.-C. Song, A blockchain and machine learning-based drug supply chain management and recommendation system for smart pharmaceutical industry, Electronics 9 (5) (2020) 852–883.
[9] W. Ahmad, A. Rasool, A.R. Javed, T. Baker, Z. Jalil, Cyber security in IoT-based cloud computing: A comprehensive survey, Electronics 11 (2021) 16–50.
[10] S. Kumar, A.K. Pundir, Blockchain—Internet of Things (IoT) enabled pharmaceutical supply chain for COVID-19, in: Proceedings of the NA International Conference on Industrial Engineering and Operations Management Detroit, Detroit, MI, USA, 2020, pp. 10–14.
[11] A.K. Bapatla, S.P. Mohanty, E. Kougianos, D. Puthal, A. Bapatla, PharmaChain: A blockchain to ensure counterfeit-free pharmaceutical supply chain, IET Netw. 12 (2) (2023) 53–76.
[12] G. Subramanian, A.S. Thampy, N.V. Ugwuoke, B. Ramnani, Crypto pharmacy–digital medicine: A mobile application integrated with hybrid blockchain to tackle the issues in pharma supply chain, IEEE Open J. Comput. Soc. 2 (2021) 26–37.
[13] Y. Yu, L. Shumei, P. Lep Yeoh, B. Vucetic, Y. Li, LayerChain : A hierarchical edge-cloud blockchain for large-scale low - delay industrial internet of things applications, IEEE Trans. Ind. Inform. 17 (2021) 5077–5086.
[14] Q. Song, Y. Chen, Y. Zhong, K. Lan, S. Fong, Z.I. of Advanced Technology of the Chinese Academy of Sciences ZIAT DACC Laboratory, F. of Management, K.U.o.S. economics, T.R. Tang, A supply-chain system framework based on internet of things using blockchain technology, ACM Trans. Internet Technol. 21 (2021) 13:1–13:24.
[15] U. Agarwal, V. Rishiwal, S. Tanwar, R. Chaudhary, G. Sharma, P.N. Bokoro, R. Sharma, Blockchain technology for secure supply chain management: A comprehensive review, IEEE Access 10 (2022) 85493–85517.
[16] Synopsis, Blockchain, 2023, https://www.synopsys.com/glossary/what-is-blockchain.html.
[17] H. Huang, J. Lin, B. Zheng, Z. Zheng, J. Bian, When blockchain meets distributed file systems: An overview, challenges, and open issues, IEEE Access 8 (2020) 50574–50586.
[18] K.R. Venugopal, N. Mangala, B.E. Reddy, An Integrated Computing Storage System and Method of Security Related Thereto, Indian Patent Application No. 202341088532, 2024.
[19] M.M. Optical Zeitgeist Laboratory, Context and selfawareness in fog and mist computing, 2017.
[20] W. R. Bezerra, F. Koch, C. Westphall, Models of Computing as a Service and IoT: an analysis of the current scenario with applications using LPWAN, Rev. Sistemas Inf. FSMA 1 (25) (2020) 56–65.
[21] J. Preden, Thinnect, Evolving computing architectures: Mist computing - IoT on the edge, 2023, https://thinnect.com/mist-computing-2/mist-computing-edge-computing-efficient-sensing-smart-device-management.
[22] K.R. Venugopal, E.E. Rajan, P.S. Kumar, Impact of wavelength converters in wavelength routed all-optical networks, Comput. Commun. 22 (3) (1999) 244–257.
[23] K.R. Venugopal, E.E. Rajan, P.S. Kumar, Performance analysis of wavelength converters in WDM wavelength routed optical networks, in: Proceedings. Fifth International Conference on High Performance Computing, IEEE, 1998, pp. 239–246.
[24] S. Tarannum, B. Aravinda, L. Nalini, K.R. Venugopal, L.M. Patnaik, Routing protocol for lifetime maximization of wireless sensor networks, Int. J. Inf. Process. 1 (1) (2007) 61–70.
[25] A. Kanavalli, D. Sserubiri, P.D. Shenoy, K.R. Venugopal, P.L. M, A flat routing protocol for sensor networks, in: 2009 Proceeding of International Conference on Methods and Models in Computer Science, ICM2CS, IEEE, 2009, pp. 1–5.
[26] Wikipedia, Edge computing, https://wikipedia.org/wiki/Edge_computing.
[27] N. Agrawal, Dynamic load balancing assisted optimized access control mechanism for Edge-Fog-Cloud network in Internet of Things environment, in: Concurrency and Computation Practice and Experience, Wiley Online Library, 2021, pp. 6440–6455.
[28] H.A. Kumar, R. J, R. Shetty, S. Roy, D. Sitaram, Comparison of IoT architectures using a smart city benchmark, Procedia Comput. Sci., Elsevier 171 (2020) 1507–1516.
[29] M. Farhadi, D. Miorandi, G. Pierre, Blockchain enabled fog structure to provide data security in IoT applications, ACM Trans. (2018).
[30] P. Mell, T. Grance, The NIST Definition of Cloud Computing, National Institute of Standards and Technology, 2011, Special Publication 800-145.
[31] H. Tianfield, Cloud computing architectures, in: 2011 IEEE Intl. Conf. on Systems, Man, and Cybernetics, 2011, pp. 1394–1399.
[32] C. Qiu, H. Yao, C. Jiang, S. Guo, F. Xu, Cloud computing assisted blockchain-enabled internet of things, IEEE Trans. Cloud Comput. 10 (1) (2022) 247–257.
[33] P. Bhattacharya, S. Tanwar, U. Bodkhe, S. Tyagi, N. Kumar, BinDaaS: Blockchain-based deep-learning as-a-service in healthcare 4.0 applications, IEEE Trans. Netw. Sci. Eng. 8 (2) (2021) 1242–1255.
[34] A. Ghadge, M. Bourlakis, S. Kamble, S. Seuring, Blockchain implementation in pharmaceutical supply chains : A review and conceptual framework, Int. J. Prod. Res. (2022) 1–19.
[35] A.P. Singh, N.R. Pradhan, A.K. Luhach, S. Agnihotri, N.Z. Jhanjhi, S. Verma, U. Ghosh, D.S. Roy, et al., A novel patient-centric architectural framework for blockchain-enabled healthcare applications, IEEE Trans. Ind. Inform. 17 (8) (2020) 5779–5789.
[36] Y. Mengmeng, Z. Tianqinng, L. Kaitai, Z. Wanlei, R.H. Deng, A blockchain-based location privacy-preserving crowdsensing system, Future Gener. Comput. Syst. 94 (2018) 408–4018.
[37] K. Randhir, K. Prabhat, T. Rakesh, G.P. Gupta, G. Sahil, M.M. Hassan, A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT network, J. Parallel Distrib. Comput. 164 (2022) 55–68.
[38] K.C. Okafor, I.E. Achumba, G.A. Chukwudebe, G.C. Ononiwu, Leveraging fog computing for scalable IoT datacenter using spine-leaf network topology, J. Electr. Comput. Eng. 2017 (2017) 1–12.
[39] K.C. Okafor, Dynamic reliability modeling of cyber-physical edge computing network, Int. J. Comput. Appl. 43 (7) (2021) 612–622.
[40] K.C. Okafor, O.M. Longe, Smart deployment of IoT-TelosB service care StreamRobot using software-defined reliability optimisation design, Heliyon 8 (6) (2022).
[41] K.C. Okafor, B. Adebisi, K. Anoh, Lightweight multi-hop routing protocol for resource optimisation in edge computing networks, Internet Things 22 (2023) 100758.

[42] L. Yuan, Z. Chuang, Y. Yu, Z. Xin, T. Zhihong, Z. Jie, A semi - centralized trust management model based on blockchain for data exchange in IoT system, IEEE Trans. Serv. Comput. (2022) 1–14.
[43] L. Tan, N. Shi, K. Yu, M. Aloqaily, Y. Jararweh, A blockchain-empowered access control framework for smart devices in green internet of things, ACM Trans. Internet Technol. (TOIT) 21 (3) (2021) 1–20.
[44] C. Xingjuan, G. Shaojin, Z. Jingbo, W. Di, C. Zhihua, Z. Wensheng, C. Jinjun, A sharding scheme-based many-objective optimization algorithm for enhancing security in blockchain-enabled industrial internet of things, IEEE Trans. Ind. Inform. 17 (2021) 7650–7658.
[45] I.A. Ridhawi, M. Aloqaily, J.Y. Jararweh, An incentive-based mechanism for volunteer computing using blockchain, ACM Trans. Internet Technol. 21 (2021) 87:1–87:22.
[46] A. Kebira, O. Ouail, S.J. Andaloussi, Access control and privacy -preserving blockchain based system for diseases management, IEEE Trans. Comput. Soc. Syst. (2022) 1–13.
[47] J. Zhang, Y. Yang, X. Liu, J. Ma, An efficient blockchain-based hierarchical data sharing for healthcare internet of things, IEEE Trans. Ind. Inform. 18 (2022) 7139–7150.
[48] J. Li, Y. Zhang, J. Ning, X. Huang, G.S. Poh, D. Wang, Attribute based encryption with privacy protection and accountability for CloudIoT, IEEE Trans. Cloud Comput. 10 (2022) 762–773.
[49] K. Yu, T. Liang, M. Aloqaily, H. Yang, Y. Jararweh, Blockchain-enhanced data sharing with traceable and direct revocation in IIOT, IEEE Trans. Ind. Inform. (2019) 1–10.
[50] S. Zawar, U. Imdad, H. Li, A. Levula, K. Khurshid, Blockchain based solutions to mitigate distributed denial of service(DDoS) attacks in the Internet of Things(IoT): A survey, MDPI 22 (2022) 1–26.
[51] M. Gunasekaran, A. Mamoun, P.M. Shakeel, C.H. Hsu, Blockchain assisted secure data sharing model for internet of things based smart industries, IEEE Trans. Reliab. 71 (2022) 348–358.
[52] Y. Zhang, C. Xu, N. Cheng, H. Li, H. Yang, X. Shen, Chronos: An accurate blockchain-based time-stamping scheme for cloud storage, IEEE Trans. Serv. Comput. 13 (2) (2019) 216–229.
[53] T. Hewa, A. Braeken, M. Liyanage, M. Ylianttila, Fog computing and blockchain-based security service architecture for 5G industrial IoT-enabled cloud manufacturing, IEEE Trans. Ind. Inform. 18 (10) (2022) 7174–7185.
[54] N. Mangala, K. Venugopal, et al., Light weight Circular Error Learning Algorithm (CELA) for secure data communication protocol in IoT-cloud systems, Int. J. Adv. Comput. Sci. Appl. 14 (7) (2023).
[55] M.M. Kermani, S. Bayat-Sarmadi, A.-B. Ackie, R. Azarderakhsh, High-performance fault diagnosis schemes for efficient hash algorithm blake, in: 2019 IEEE 10th Latin American Symposium on Circuits & Systems, LASCAS, IEEE, 2019, pp. 201–204.
[56] A.C. Canto, M.M. Kermani, R. Azarderakhsh, CRC-based error detection constructions for FLT and ITA finite field inversions over GF (2 m), IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 29 (5) (2021) 1033–1037.
[57] A.C. Canto, A. Sarker, J. Kaur, M.M. Kermani, R. Azarderakhsh, Error detection schemes assessed on FPGA for multipliers in lattice-based key encapsulation mechanisms in post-quantum cryptography, IEEE Trans. Emerg. Top. Comput. (2022).
[58] J. Kaur, A.C. Canto, M.M. Kermani, R. Azarderakhsh, Hardware constructions for error detection in WG-29 stream cipher benchmarked on FPGA, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. (2023).
[59] M.M. Kermani, Fault Detection Schemes for High Performance Vlsi Implementations of the Advanced Encryption Standard (Ph.D. thesis), Faculty of Graduate Studies, University of Western Ontario, 2007.
[60] M.M. Kermani, R. Azarderakhsh, M. Mirakhorli, Multidisciplinary approaches and challenges in integrating emerging medical devices security research and education, in: 2016 ASEE Annual Conference & Exposition, 2016.
[61] M. Mozaffari-Kermani, R. Azarderakhsh, K. Ren, J.-L. Beuchat, Guest editorial: introduction to the special section on emerging security trends for biomedical computations, devices, and infrastructures, IEEE/ACM Trans. Comput. Biol. Bioinform. 13 (03) (2016) 399–400.
[62] A.C. Canto, M.M. Kermani, R. Azarderakhsh, Reliable constructions for the key generator of code-based post-quantum cryptosystems on FPGA, ACM J Emerg. Technol. Comput. Syst. 19 (1) (2022) 1–20.
[63] B. Koziel, A. Jalali, R. Azarderakhsh, D. Jao, M. Mozaffari-Kermani, NEON-SIDH: Efficient implementation of supersingular isogeny Diffie-Hellman key exchange protocol on ARM, in: Cryptology and Network Security: 15th International Conference, CANS 2016, Milan, Italy, November 14-16, 2016, Proceedings 15, Springer, 2016, pp. 88–103.
[64] M.B. Niasar, R. Azarderakhsh, M.M. Kermani, Optimized architectures for elliptic curve cryptography over Curve448, Cryptol. ePrint Arch. (2020).
[65] A. Cintas-Canto, M.M. Kermani, R. Azarderakhsh, Reliable architectures for finite field multipliers using cyclic codes on FPGA utilized in classic and post-quantum cryptography, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 31 (1) (2022) 157–161.
[66] R.A. Karam, S. Katkoori, M.M. Kermani, Work-in-progress: Hyflex hands-on hardware security education during covid-19, in: 2022 IEEE World Engineering Education Conference, EDUNINE, IEEE, 2022, pp. 1–4.
[67] D. Lin, J. Wu, Q. Yuan, Z. Zheng, Modeling and understanding ethereum transaction records via a complex network approach, IEEE Trans. Circuits Syst. II 67 (11) (2020) 2737–2741.
[68] J. Eberhardt, S. Tai, On or off the blockchain? Insights on off-chaining computation and data, in: Service-Oriented and Cloud Computing: 6th IFIP WG 2.14 European Conference, ESOCC 2017, Oslo, Norway, September 27-29, 2017, Proceedings 6, Springer, 2017, pp. 3–15.