International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 1s
DOI: https://doi.org/10.17762/ijritcc.v10i1s.5824
***Article Received**: 30 August 2022 **Revised**: 02 November 2022 **Accepted**: 28 November 2022*

SCMBQA: Design of a Customised SCM-Aware Sidechaining Model for QoS Enhancement under Attack Scenarios

Sonali V. Shinkar[a,*], Dr.Dolly Thankachan[b]
[a]Research Scholar Oriental University, Indore.
[b]Associate Professor & HOD, Oriental University, Indore
*Corresponding author *E-mail: shinkar.sonali15@gmail.com

Abstract
Storing& processing data for supply chain management (SCM) systems requires design of high-security and quality of service (QoS) aware models. These modelsassist in improving traceability performance of SCM deployments via integration of transparent & distributed mechanisms. A wide variety of security models are proposed by researchers to perform these tasks, and it is observed that blockchain-based SCM implementations outperform other models in terms of security & QoS metrics.But most of these implementationsare general-purpose and do not incorporate SCM-specific consensus & mining rules. It is also observed that, mining speed& throughput performance of these blockchain-based implementations reduces exponentially w.r.t. number of SCM transactions. To resolve these issues, this paper discusses design of a novel Proof-of-Supply Chain (PoSC) based consensus model, which is specifically designed for sidechain based SCM deployments. The PoSC consensus model is used for high-efficiency SCM-based data storage and communication scenarios. The proposed PoSC consensus model is capable of resisting selfish mining, time jacking, and sybil attacks, which are targeted towards SCM deployments. The model uses temporal performance metrics of miner nodes, and combines them with relationship graphs to form an SCM miner rank. Based on this rank, miner nodes are selected, and their consensus responses are recorded. These responses are processed using an augmented deep learning model, that is trained over 8 different SCM implementations via machine learning. After successful mining, responses obtained from these miners are used to incrementally train the machine learning model which assists in continuous performance improvement. The SCMBQA model was tested on milk supply chain, agriculture supply chain, and electronic supply chain applications, in terms of computational speed, throughput, energy requirement, retrieval & verification delay, and storage requirements. It was observed that the proposed PoSC consensus was capable of improving the computational speed by 8.5%, reduce energy consumption by 4.9%, improve throughput by 9.6%, and reduce storage costs by 15.4% when compared with standard blockchain-based SCM consensus models. This is because the proposed model deploys an intelligent sidechaining approach, that is capable of optimizing number of generated sidechains via temporal QoS & security performance metrics. Due to use of smaller chain lengths, the proposed model is capable of integrating privacy-aware & secure approaches depending upon different SCM stages. Thus, distributor-level security models are different than retailer-level security models, which assists in context-sensitive block deployments. Due to use of PoSC, the proposed model was observed to be 99.5% resilient against internal and external attacks, which makes it useful for real-time SCM deployments.

Keywords: Blockchain, sidechain, context, SCM, security, privacy, QoS, consensus

1. Introduction

Design of supply chain management (SCM) models is a multidomain task, which incorporates data management, tracking, entity-based storage, security deployment, and data validations. SCM data should be immutable, which ensures that data, once written should not be modified or removed. The stored data should be transparent, thus enabling entity-based access for improved trust. It should be traceable, thus ensuring that access to temporal data should be delay-aware, and tamper-proof. The data should also be processed in a secure & distributed manner, which ensures better QoS performance, and assists multiple trusted entities to incorporate progress updates. All these characteristics are present in blockchain-based models, which makes them highly useful for deploying SCM applications [1]. A typical blockchain-based SCM model is depicted in figure 1, wherein different components including supplier, manufacturer, distributor, and retailer are connected via different smart contract-based blockchains [2].It can be observed that supplier & manufacturer are linked via smart contracts, and information related to raw materials & manufacturing details are stored on the blockchain. Another smart contract is used to store retailer & product information along with inventory status, which assists in tracking product-level sale, purchase, and procurement details. While adding a block to SCM blockchains, a number of delay components are encountered.

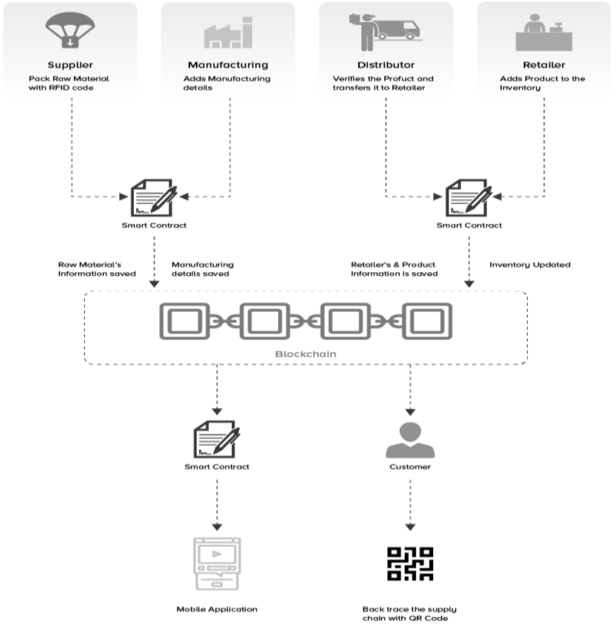
Supplier
Pack Raw Material with RFID code
Manufacturing
Adds Manufacturing details
Distributor
Verifies the Profuct and transfers it to Retailer
Retailer
Adds Product to the Inventory
Smart Contract
Smart Contract
Raw Material's Information saved
Manufacturing details saved
Retailer's & Product Information is saved
Inventory Updated
Blockchain
Smart Contract
Customer
Mobile Application
Back trace the supply chain with QR Code

Figure 1. A typical blockchain based SCM model

These components can be observed via equation 1, wherein block reading delay, writing delay, verification delay, and hashing delay are combined with different SCM operations.

$$D(mining) = \left| \sum_{i=1}^{N_{rounds}} N * D_{read} + (N-1) * (D_{validate} + D_{hash}) + D_{write} \right|^{N_{SCM}} \quad \dots (1)$$

Where, $N_{rounds}, \& N_{SCM}$ indicates number of rounds needed to verify the block, and number of SCM operations needed to create a block, while N represents number of blocks currently present in the blockchain. Based on this equation, it can be observed that delay needed to add a block to the chain is directly proportional to chain length, thus, as length of chain increases, the delay needed to mine/add these blocks will also increase. To reduce this delay, various sidechaining based SCM models are proposed by researchers. Survey of these models [3,4,5] along with their nuances, advantages, limitations, and future research scopes is discussed in the next section of this text. Based on this discussion, it can be observed that very few blockchain-based models are specifically designed to handle SCM-specific issues, which limits their QoS & security performance when applied to large-scale SCM deployments. To overcome this drawback, section 3 proposes design of a novel customised SCM-aware sidechaining model for QoS enhancement under attack scenarios. The proposed model divides central SCM blockchain into different sub-parts using a bioinspired Genetic Algorithm (GA), which optimizes number of chains, and individual chain lengths. These smaller chains are customized depending upon individual SCM entities, wherein their security & privacy parameters are tuned for attack resistance in real-time scenarios. The proposed model's performance was evaluated in terms of computational speed, throughput, energy requirement, retrieval & verification delay, & storage requirements in section 4, and compared with various state-of-the-art models. Based on this performance, researchers will be able to validate SCMBQA's performance, and identify any performance issues which can be resolved via model-based optimizations. Finally, this text concludes with some interesting observations about the proposed model, and recommends methods to further improve its performance.

2. Literature Review

A wide variety of SCM models utilize blockchain-based solutions to improve their underlying security & QoS performance. For instance, work in [4, 5, 6] proposes blockchain Model for Agri-Food SCM, vendor Management &inventory SCM (VMI SCM), and Soybean traceability using blockchain based SCM models. These models are highly application-specific, and cannot be applied to general purpose SCM applications. Recommendations to improve blockchain scalability for SCM are discussed in [7], wherein sidechaining, deep learning, and other machine learning based models are proposed. These models are further reviewed in [8, 9, 10], wherein blockchain for logistics, general purpose blockchains for SCM, and healthcare SCM using smart contracts (SC)are discussed. Efficiency of these models is further enhanced in terms of delay, and throughput performance via the work in [11, 12, 13], where researchers have proposed simplified user interfaces, strategic product deployments, and decentralized digital manufacturing during CoVID (DDM SCM) are discussed. These models aim at reducing redundancies during mining for enhancement of mining speed, with lower energy consumption for different applications.

Blockchain-based SCM systems are also used for enterprise [14], drug traceability [15], general purpose deployments [16], pharma tracking [17], and agricultural food supply tracking via reinforcement learning & safety management [18, 19, 20] with highly efficient deployments. A combination of these models is discussed in [21], wherein distributed ledger technology (DLT) based tagging is performed to improve overall SCM performance. Extended applications of blockchains for SCM are further discussed in [22, 23, 24], wherein technical products SCM, data sanitization based SCM, and enhanced logistics based SCM are discussed in details. These models have better security and QoS performance, but cannot be used for multiple SCM applications, which limits their scalability. To improve this scalability, work in [25, 26, 27] proposes use of machine learning, Economic SCM, and product traceability with high throughput via distributed computing models. These approaches are further explored in [28, 29, 30], wherein challenges with blockchain based SCM models, reviews from product managers about blockchain based SCMs, and their applications to different food industries are discussed in details. Based on these approaches, work in [31, 32, 33] propose sustainable models for automotive industry, CoVID-19 based SCM applications, and Governance design of blockchain-based consortia are proposed by researchers. These models aim at reducing redundancies in application-specific SCM deployments. Other models that reduce managing risks of blockchain for SCM [34], alternatives to blockchain for SCM [35], and factors needed for adoption & accpetance of blockchain in SCM [36] are discussed. Based on these discussions, it can be observed that none of the existing approaches proposed SCM-specific consensus & blockchain models, which limits scalability of SCM application deployments. To improve this scalability, while maintaining high security, next section proposes a customised SCM-aware sidechaining model for QoS enhancement under attack scenarios. Performnace of the proposed model was also evaluated under different network scenarios, and compared with various state-of-the-art blockchain based SCM models for validation of the proposed model, and its applicability under different attack scenarios.

3. Design of acustomised SCM-aware sidechaining model for QoS enhancement under attack scenarios

From the literature survey, it can be observed that a wide variety of blockchain models are proposed for design of SCM based deployments. These models utilize general purpose consensus methods including Proof-of-Work (PoW), Proof-of-Stake (PoS), Practical Byzantine Fault Tolerance (PBFT), etc. which reduces their scalability when applied to real-time SCM applications. This is because SCM-based deployments require entity-specific privacy & security models that must be tuned as per instantaneous scenario requirements. Moreover, performance of existing blockchain-based SCM models reduces exponentially w.r.t. length of the blockchain, which further limits their deployment capabilities. To overcome these drawbacks, this section proposes design of SCMBQA, which is a customised SCM-aware sidechaining model for QoS enhancement under attack scenarios. Overall flow of the proposed model is depicted in figure 2, wherein creation of sidechain along with SCM based Proof-of-Supply Chain (PoSC) consensus method is visualized.From this flow it can be observed that, the proposed model is built using the following components,

- An entity-level analysis block, which is capable of evaluating context-sensitive parameters like delay, security level, and other QoS metrics
- A miner selection layer, which assists in identification of miner rank, based on entity & miner relationship
- A Proof-of-Supply Chain (PoSC) block that enables SCM based consensus
- Deep learning & incremental learning models, that assist in sidechain formation and performance tuning via Q-learning based decisions

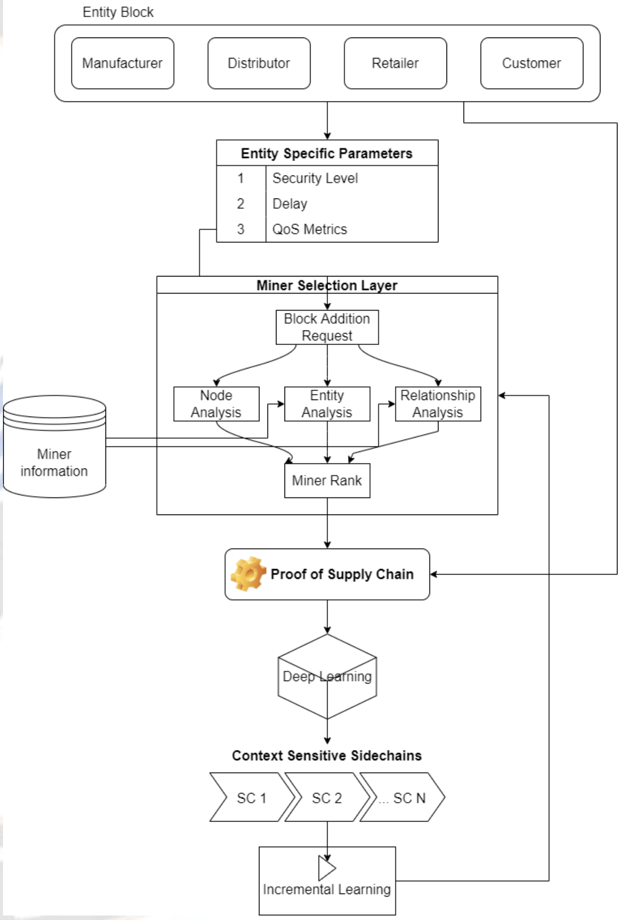

Figure 2. Overall flow of the proposed SCM-based sidechaining model

Design of these blocks, along with their internal processes are discussed in different sub-sections of this text. Researchers can implement the entire model is part(s) or as a whole after referring these sub-sections.

3.1. Design of entity-level analysis block, for evaluating context-sensitive parameters

A typical SCM model is built up of multiple entities, which include but are not limited to, product manufacturer, distributor, retailer, and customer. Each of these entities require a different level of security & QoS performance. For each entity, security level (SL) is evaluated via equation 2,

$$SL_i = \sum_{l=1}^{N_{i_a}} P_l * \frac{\log\left(\frac{1}{P_l}\right)}{N_{i_a}} \dots (2)$$

Where, N_{i_a} represents number of attacks prevalent for the given entity, while P_l represents probability of that attack. This attack probability is evaluated via equation 3, by reviewing different attack models, and estimating their effect on SCM performance.

$$P_l = \sum_{i=1}^{N_{SCM}} \frac{N(blocks)_{invalid}}{N(blocks)_{total}} * \frac{1}{N_{SCM}} \dots (3)$$

Where, N_{SCM} represents total number of SCM models used for validation, $N(blocks)_{total}$& $N(blocks)_{invalid}$ represents total blocks in the chain, and number of blocks invalidated due to given attack type. Similarly, QoS level for a given entity is evaluated via equation 4, wherein delay requirements, energy requirements, and throughput requirements of the entity are considered.

$$QoS_i = \left(\sum_{j=1}^{NP} \frac{E_j}{Max(E)} + \frac{D_j}{Max(D)} + \frac{Max(TH)}{TH_j}\right) * \frac{1}{NP} \dots (4)$$

Where, $E, D, and\ TH$ represents energy requirement, delay requirement, and throughput requirement for a given entity, while NP represents total number of SCM models previously processed for estimating these values. Based on temporal evaluation from different SCM deployments, security & QoS levels are estimated, and given to the miner selection layer. Design of this layer is discussed in the next sub-section of this text.

3.2. Design of miner selection layer for miner rank evaluation, based on entity & miner relationship

This layer evaluates miner rank based on SCM entity, and its relationship with current set of miners. In this layer, parameters includingentityinformation, temporal mining validations, mining performance, and approximate energy requirements for mining blocks are used for evaluation of intermediate miner rank. This rank is fine-tuned via relationship estimation between miner node and current entity parameters.These parameters includeminer node to entity distance, and temporal probability of miner to minecurrent entityblocks. The information collected during this phase is stored on edge nodes, for faster analysis & future retrieval purposes. The miner selection mode worksvia the following process,

- Initialize model parameters including,
  - Total number of blocks in current sidechain = $N_{current}$
  - Location of node from which block request is originating(s_x, s_y), andlocation of miner nodes (M_x, M_y)
- Evaluate mining performance for each miner, using the following metrics,
  - Total number of correctly mined blocks by the miner ($N_{correct}$) is evaluated via equation 5,

$$N_{correct} = \sum_{i=1}^{N_{total}} |PM_i = 1| \dots (5)$$

Where, $PM$is probability of mining the block correctly, and N_{total} represents total number of blocks mined by the miner.

- Average mining delay for the current miner is evaluated via equation 6,

$$AVG(Delay) = \frac{1}{N_{current}} \sum_{i=1}^{N_{correct}} Mine(Delay_i) \dots (6)$$

Where, $AVG(Delay), and\ Mine(Delay)$ represents average mining delay, and total mining delay for correctly mined blocks.

- Energy needed to mine the blocks by this miner is evaluated via equation 7 as follows,

$$AVG(Energy) = \frac{1}{N_{correct}} \sum_{i=1}^{N_{correct}} Mine(Energy_i) \dots (7)$$

Where, $AVG(Energy), and\ Mine(Energy)$ represents average mining energy, and total mining energy for correctly mined blocks.

- For the entity that is requesting to add a block to the chain, following metrics are evaluated,
  - Distance between the entity and miner node is evaluated via equation 8,

$$D(M,S) = \sqrt{(M_x - s_x)^2 + \left(M_y - s_y\right)^2} \dots (8)$$

- Probability with which the current miner node is temporally mining blocks from current entity is evaluated via equation 9,

$$P(M,S) = \frac{\sum_{i=1}^{N_{correct}} B(M,S)}{N_{correct}} \dots (9)$$

Where, $B, \& P$ represents number of blocks originated by this entity which are mined by current miner node, and probability of miner to source relationship. These metrics are combined to form the final miner rank via equation 10,

$$M_{rank_i} = [\left(1 - \frac{AVG(Delay_i)}{Max\left(\bigcup_{j=1}^{N_{correct}} Delay_j\right)}\right) +$$

***Article Received**: 30 August 2022 **Revised**: 02 November 2022 **Accepted**: 28 November 2022*

$$\frac{N_{correct_i}}{N_{total}} + \frac{N_x + N_y - D(M_i, S)}{N_x + N_y} + \left(1 - \frac{AVG(Energy_i)}{Max\left(\bigcup_{j=1}^{N_{correct}} Energy_j\right)}\right) + \left(1 - P(M_i, S)\right)] * \frac{Old(M_{rank})}{Max[\bigcup_{j-1}^{N(Miners)} Old(M_{rank_j})]} \dots (10)$$

Where, $N_x, \& N_y$ represent X & Y dimensions of the network, $Max(Delay), \& \, Max(Energy)$ represents maximum delay incurred during mining, and maximum energy needed for mining a block from the total number of blocks currently mined in the sidechain, $Old(M_{rank})$ represents old miner ranks which are initialized to ‘1’, and later modified via Q-learning process. These ranks are evaluated for each miner node, and a rank threshold is evaluated via equation 11,

$$M_{rank_{th}} = \sum_{i=1}^{Num(Miners)} \frac{M_{rank_i} * N_{fields}}{N_{f_{total}} * Num(Miners)} \dots (11)$$

Where, $N_{fields} \& N_{f_{total}}$ represents total number of modifiable fields in the current block structure which assist in unique identification of hash values, and total number of fields in the block structure, while$Num(Miners)$ represents total number of miners in the current blockchain network deployment. All miners that have a rank above threshold are selected for final mining process, while others are not used for current consensus. Responses of these miners are processed via a PoSC based consensus model which is described in the next section of this text.

3.3. Design of Proof-of-Supply Chain (PoSC) that enables SCM based consensus

Once miners are selected, then their temporal security performance is evaluated to form PoSC based consensus. To evaluate this performance, following process is followed,

- For each selected miner node, perform N dummy attacks, and evaluate its security level via equation 12,

$$I_{sl} = \frac{\sum_{i=1}^{N} \sum_{j=2}^{N_{blocks}} Hash_{j-1} = Hash_{prev_j}}{N * N_{blocks}} \dots (12)$$

Where, $Hash, \& \, Hash_{prev}$ represents hash of current block, and previous hash of current block. Both these values assist in verification of current sidechain. Once these values are evaluated, revert the blockchain of current miner node to its pre-attack state to maintain validity.

- Estimate temporal attack resilience of the miner via equation 13,

$$T_{AR} = \frac{\sum_{i=1}^{N_{attacks}} I_{sl}(i)}{N_{attacks}} \dots (13)$$

Where, $N_{attacks}$ represents number of attacks previously identified by the miner node for current sidechain blocks.

- Evaluate the final miner score for selected miners via equation 14 as follows,

$$M_{final} = M_{rank} * T_{AR} * I_{sl} \dots (14)$$

Miners, where M_{final} is positive are selected for mining, while other miner nodes are discarded from the mining process. This ensures that miners with validated sidechains are used for mining new blocks. Miner nodes with value of $M_{final} = 0$, are reported to network administrator, which will assist in validation and correction of their internal blockchains.

3.4. Design of deep learning & incremental learning models for sidechain management

Once miner nodes are selected, and blocks are now ready to be added to the blockchain, a novel deep learning model that uses Genetic Algorithm (GA) with Q-learning is used for sidechain management. This model is activated for each block addition request, and works via the following process,

- Initialize GA parameters, which include,

○ Number of iterations (N_i)

○ Number of solutions (N_s)

○ Learning rate (L_r)

○ Number of sidechains currently present (N_{sc})

- Initially mark all solutions as ‘to be modified’
- For each iteration in 1 to N_i

○ For each solution in 1 to N_s

▪ If this solution is marked as ‘not to be modified’, then go to the next solution

▪ Else, generate a new solution via the following process,

- Select a random chain from the list of chains via equation 15,

$$C_{select} = random(1, N_{sc}) \dots (15)$$

- Evaluate solution fitness, which combines average delay & energy needed for adding N dummy transactions in the chain via equation 16,

$$fitness = \sum_{i=1}^{N} \frac{D_i(C_{select})}{N_{sc} * MAX(D)} + \frac{E_i(C_{select})}{N_{sc} * Max(E)} + \frac{Max(TH)}{N_{sc} * TH_i} \dots (16)$$

Where, $D, E, and \, TH$represents average delay of adding new blocks, energy needed to add new blocks, and throughput obtained after adding new block into the current sidechain.

- If this fitness is more than current average fitness of all solutions, then it is discarded, and solution is marked as ‘to be modified’

▪ Count total number of solutions per iteration which are marked as ‘to be modified’ (T_{s_i}) via equation 17,

$$T_{s_i} = \sum_{i=1}^{N_s} |S_i == Modify| \dots (17)$$

- ○ Repeat this for all iterations, and identify value of T_{s_i} for the last iteration
- If at the final iteration, T_{s_i} is more than solution threshold (T_{sol}), then a new sidechain is generated, else chain with lowest number of blocks is used for adding new blocks into the system. Value of T_{sol} is evaluated via equation 18 as follows,

$$T_{sol} = L_r * \frac{\sum_{i=1}^{N_s} T_{s_i}}{N_s} \dots (18)$$

The value of F_{fact} is selected via equation 19, and it assists in creating an optimum number of sidechains for the current SCM model.

$$L_r = L_{r_{old}} + \frac{\left[\begin{array}{c} Max\left(\bigcup_{i=1}^{N_{sc}} L_i\right) * L_{r_{old}} \\ -AVG\left(\bigcup_{i=1}^{N_{sc}} L_i\right) \end{array} \right]}{Max\left(\bigcup_{i=1}^{N_{sc}} L_i\right)} \dots (19)$$

Where, L_i represents length of chain for the i^{th} sidechain. Due to use of previous learning rate, equation 19 can reduce number of unnecessary sidechains, thereby reducing delay needed for sidechain management. Hashes for each block are stored on a separate chain, along with sidechain ID, which assists in fast retrieval of data whenever necessary.Blocks are added to the chain via PoSC consensus, which assists in reducing probability of attacks in the SCM network. Once a batch of blocks are added to the sidechain, then evaluation of average block addition delay, energy required by miners, and throughput are calculated. Based on these parameters, miner rank is modified via equation 20 that implements Q-learning as follows,

$$New(M_{rank}) = Old(M_{rank}) + L_r * [fitness + P_l * \max(M_{rank}) - Old(M_{rank})] \dots (20)$$

This new rank metric is used to interpolate miner performance, and assists in selection of miners for next set of block addition requests. Evaluation of the proposed model in terms of computational speed, throughput, energy requirement, retrieval & verification delay, & storage requirements under different types of attacks is discussed in the next section of this text.

4. Results and comparative evaluation

The proposed SCMBQA model evaluates a miner rank, and combines it with Q-learning and GA models for improving the efficiency of miner selection in sidechain-based SCM deployments. The proposed model also uses a PoSC based consensus to further improve overall mining performance. To evaluate parametric performance of the proposed model, SCM dataset from Auto Supply Chain Data(https://data.mendeley.com/datasets/n24z7r2z28/2/files/89b1d403-2c23-41e8-888b-793ac998dd0d), SCM dataset from Brunel University (https://brunel.figshare.com/articles/dataset/Supply_Chain_Logistics_Problem_Dataset/7558679), Global Garment dataset from Datahub (https://old.datahub.io/dataset/global-garment-supply-chain-data), and Smart Supply Chain dataset from Kaggle (https://www.kaggle.com/shashwatwork/dataco-smart-supply-chain-for-big-data-analysis), were used. Each of these datasets had 750k blockchain transactions, out of which 75% were used for training the model, while the remaining 25% were used for model evaluation. Based on this strategy, average mining delay (D), energy consumed during mining (E), mining throughput in blocks mined per minute (T) & storage costs (S) were evaluated, and compared with the mining optimization models proposed in VMI SCM [5], SC [10], and DDM SCM [13]. All these evaluations were performed under different types of attacks, which assists in estimating real-time performance of the SCM deployments.

Evaluation of mining delay is tabulated w.r.t. number of blocks mined (NB) in table 1, wherein SCM was deployed under selfish mining attacks. In each case, maximum 10% of all miner nodes were considered to be affected by the attack, which assists in real-time performance estimation of the proposed model under different attack types.

Table 1. Average mining delay for Selfish Mining Attack w.r.t. number of blocks in the blockchain

NB	**D (s)** **VMI** **SCM [5]**	**D (s)** **SC** **[10]**	**D (s)** **DDM** **SCM [13]**	**D (s)** **SCMB** **QA**
24k	1.74	2.00	1.99	1.19
50k	2.88	3.24	3.29	1.98
75k	4.68	4.86	5.35	3.21
100k	8.01	7.80	9.15	5.49
125k	12.48	11.44	14.26	8.56
150k	21.48	23.86	24.55	14.73
176k	33.06	38.25	36.01	22.43
200k	48.24	58.05	52.55	32.74
225k	51.57	56.09	56.18	35.00
300k	54.27	61.29	59.11	36.82
350k	61.03	69.01	66.47	41.41
400k	67.78	76.75	73.84	46.00
450k	74.55	84.49	81.20	50.58
500k	81.31	92.22	88.57	55.17
550k	88.07	96.72	91.77	59.21
600k	94.83	104.20	98.81	63.75

650k	101.59	111.69	105.86	68.30
675k	108.35	119.18	112.91	72.85
725k	115.12	126.67	119.95	77.39
750k	121.87	134.14	126.99	81.93

From this evaluation, it can be observed that the proposed model is 20.5% faster than VMI SCM [5], 24.3% faster than SC [10], and 20.9% faster than DDM SCM [13] for Selfish Mining Attack. This is because of optimum miner selection, and incorporation of delay while selection of miner nodes. Similarly, average mining delay for Time Jacking Attack model was evaluated, and can be observed from table 2 as follows:

Table 2. Average mining delay for Time Jacking Attack model w.r.t. number of blocks in the blockchain

NB	**D (s) VMI SCM [5]**	**D (s) SC [10]**	**D (s) DDM SCM [13]**	**D (s) SCMB QA**
24k	2.31	2.62	2.64	1.59
50k	3.79	4.05	4.32	2.59
75k	6.35	6.33	7.26	4.35
100k	10.25	9.62	11.71	7.03
125k	16.99	17.65	19.41	11.65
150k	27.27	31.05	30.28	18.58
176k	40.65	48.15	44.28	27.59
200k	49.91	57.07	54.37	33.87
225k	52.92	58.68	57.65	35.91
300k	57.65	65.15	62.80	39.12
350k	64.40	72.88	70.16	43.70
400k	71.17	80.62	77.52	48.29
450k	77.93	88.35	84.88	52.88
500k	84.69	94.47	90.17	57.19
550k	91.45	100.45	95.29	61.47
600k	98.21	107.94	102.33	66.02
650k	104.97	115.43	109.38	70.57
675k	111.73	122.92	116.43	75.11
725k	118.49	130.40	123.47	79.66
750k	125.26	142.93	137.00	85.07

From this evaluation, it can be observed that the proposed model is 18.5% faster than VMI SCM [5], 20.6% faster than SC [10], and 23.8% faster than DDM SCM [13] for Time Jacking Attack. This is because of optimum miner selection, and incorporation of delay while selection of miner nodes. Similarly, average mining delay for Sybil Attack was evaluated, and can be observed from table 3 as follows:

Table 3. Average mining delay for Sybil Attack w.r.t. number of blocks in the blockchain

NB	**D (s) VMI SCM [5]**	**D (s) SC [10]**	**D (s) DDM SCM [13]**	**D (s) SCMB QA**
24k	4.51	4.94	5.15	3.09
50k	7.50	7.69	8.57	5.14
75k	12.29	11.81	14.05	8.43
100k	20.17	20.20	23.05	13.83
125k	32.78	36.07	36.80	22.39
150k	50.31	58.66	55.23	34.19
176k	67.08	77.94	73.07	45.52
200k	76.17	85.75	82.97	51.69
225k	81.90	91.73	89.21	55.58
300k	90.41	102.25	98.49	61.35
350k	100.43	113.71	109.39	68.15
400k	110.44	125.16	120.29	74.94
450k	120.46	135.43	129.67	81.53
500k	130.47	144.39	137.38	87.90
550k	140.48	154.37	146.39	94.44
600k	150.50	165.45	156.83	101.18
650k	160.52	176.55	167.26	107.91
675k	170.53	187.64	177.70	114.65
725k	180.55	202.47	192.94	122.02
750k	190.93	215.25	205.37	129.21

From this evaluation, it can be observed that the proposed model is 18.1% faster than VMI SCM [5], 20.6% faster than SC [10], and 18.9% faster than DDM SCM [13] for Sybil Attack. This is because of optimum miner selection, and improving the mapping efficiency of capacity to blockchain mining resource requirements. Similarly, evaluation of energy needed for mining is tabulated w.r.t. number of blocks used for mining (NB) in table 4, wherein Selfish Mining Attack model was used.

Table 4. Average mining energy needed by Selfish Mining Attack w.r.t. number of blocks in the blockchain

NB	**E (mJ) VMI SCM [5]**	**E (mJ) SC [10]**	**E (mJ) DDM SCM [13]**	**E (mJ) SCMB QA**
24k	1.71	1.94	1.93	1.16
50k	2.80	3.00	3.20	1.92
75k	4.70	4.69	5.37	3.22
100k	7.59	7.13	8.67	5.21
125k	12.57	13.07	14.37	8.62
150k	20.20	23.00	22.43	13.76
176k	30.11	35.67	32.80	20.43

200k	36.97	42.27	40.27	25.08
225k	39.20	43.47	42.70	26.60
300k	42.70	48.25	46.51	28.98
350k	47.71	53.99	51.97	32.37
400k	52.72	59.72	57.43	35.77
450k	57.73	65.45	62.88	39.17
500k	62.73	69.97	66.79	42.36
550k	67.74	74.41	70.59	45.54
600k	72.75	79.96	75.81	48.91
650k	77.75	85.51	81.03	52.27
675k	82.76	91.05	86.24	55.64
725k	87.77	96.60	91.47	59.01
750k	92.78	102.46	97.09	62.43

From this evaluation, it can be observed that the proposed model has 16.6% lower energy consumption than VMI SCM [5], 20.9% lower energy consumption than SC [10], and 18.2% lower energy consumption than DDM SCM [13] for Selfish Mining Attack model. This is because of optimum miner selection, and incorporation of residual energy during selection of miner nodes. Similarly, average mining energy needed for Time Jacking Attack models was evaluated, and can be observed from table 5 as follows:

Table 5. Average mining energy needed Time Jacking Attack model w.r.t. number of blocks in the blockchain

NB	E (mJ) VMI SCM [5]	E (mJ) SC [10]	E (mJ) DDM SCM [13]	E (mJ) SCMB QA
24k	2.26	2.47	2.57	1.54
50k	3.75	3.85	4.29	2.57
75k	6.15	5.91	7.03	4.21
100k	10.09	10.10	11.53	6.92
125k	16.39	18.04	18.41	11.20
150k	25.16	29.33	27.61	17.10
176k	33.55	38.97	36.53	22.76
200k	38.09	42.87	41.48	25.84
225k	40.95	45.87	44.60	27.79
300k	45.21	51.12	49.23	30.67
350k	50.21	56.85	54.69	34.07
400k	55.22	62.58	60.15	37.47
450k	60.23	67.71	64.84	40.77
500k	65.23	72.19	68.69	43.95
550k	70.25	77.19	73.19	47.22
600k	75.25	82.73	78.41	50.59
650k	80.26	88.28	83.63	53.96
675k	85.27	93.81	88.85	57.32
725k	90.27	101.23	96.47	61.01
750k	95.47	107.63	102.69	64.61

From this evaluation, it can be observed that the proposed model has 15.8% lower energy consumption than VMI SCM [5], 22.9% lower energy consumption than SC [10], and 22.3% lower energy consumption than DDM SCM [13] for Jamming Attack. Similarly, average mining energy needed for Sybil Attacks was evaluated, and can be observed from table 6 as follows:

Table 6. Average mining energy needed Sybil Attack w.r.t. number of blocks in the blockchain

NB	E (mJ) VMI SCM [5]	E (mJ) SC [10]	E (mJ) DDM SCM [13]	E (mJ) SCMB QA
24k	1.99	2.09	2.22	1.41
50k	3.27	3.23	3.70	2.34
75k	5.31	4.93	6.07	3.83
100k	8.76	8.48	9.95	6.29
125k	14.43	15.33	15.91	10.26
150k	22.37	25.15	23.90	15.76
176k	29.83	33.43	31.62	20.97
200k	33.72	36.66	35.89	23.75
225k	36.24	39.21	38.60	25.53
300k	40.05	43.74	42.61	28.21
350k	44.49	48.65	47.33	31.33
400k	48.94	53.56	52.05	34.46
450k	53.39	58.00	56.14	37.49
500k	57.83	61.92	59.53	40.41
550k	62.27	66.25	63.47	43.42
600k	66.71	71.01	67.99	46.51
650k	71.16	75.77	72.51	49.61
675k	75.61	80.53	77.04	52.71
725k	80.05	86.77	83.56	56.12
750k	84.65	92.25	88.95	59.43

From this evaluation, it can be observed that the proposed model has 16.5% lower energy consumption than VMI SCM [5], 19.8% lower energy consumption than SC [10], and 16.8% lower energy consumption than DDM SCM [13] for Sybil Attack. Similarly, evaluation of average throughput in blocks mined per minute is tabulated w.r.t. number of blocks used for mining (NB) in table 7, wherein Selfish Mining Attack model was used.

Table 7. Average throughput for Selfish Mining Attack w.r.t. number of blocks in the blockchain

NB	**T (bpm) VMI SCM [5]**	**T (bpm) SC [10]**	**T (bpm) DDM SCM [13]**	**T (bpm) SCMB QA**
24k	177.61	159.88	159.31	264.18
50k	111.04	104.33	96.92	163.20
75k	72.07	71.47	58.49	102.71
100k	48.68	49.65	35.43	66.40
125k	29.57	29.53	21.89	40.61
150k	14.89	13.43	13.49	22.19
176k	9.92	8.36	9.19	14.87
200k	7.83	6.81	7.23	11.71
225k	7.29	6.68	6.81	10.96
300k	6.69	5.99	6.27	10.08
350k	5.99	5.35	5.60	9.01
400k	5.41	4.83	5.07	8.14
450k	4.93	4.40	4.63	7.43
500k	4.53	4.09	4.33	6.86
550k	4.30	3.84	4.11	6.40
600k	3.91	3.57	3.83	5.96
650k	3.65	3.33	3.56	5.56
675k	3.44	3.13	3.35	5.23
725k	3.25	2.96	3.16	4.94
750k	3.09	2.80	2.97	4.68

From this evaluation, it can be observed that the proposed model is 6.5% better throughput than VMI SCM [5], 8.9% better throughput than SC [10], and 7.4% better throughput than DDM SCM [13] for EHR based application model. This is because of optimum miner selection, and incorporation of delay during miner node selection process. Similarly, average throughput for Time Jacking Attack models was evaluated, and can be observed from table 8 as follows:

Table 8. Average throughput for Jamming Attack w.r.t. number of blocks in the blockchain

NB	**T (bpm) VMI SCM [5]**	**T (bpm) SC [10]**	**T (bpm) DDM SCM [13]**	**T (bpm) SCMB QA**
24k	136.17	126.04	120.53	201.36
50k	86.87	84.63	72.95	125.57
75k	57.80	58.43	44.09	80.32
100k	35.73	36.05	27.04	49.49
125k	19.13	18.11	16.67	28.04
150k	11.83	10.28	10.93	17.72
176k	8.64	7.39	8.04	12.97
200k	7.49	6.72	7.01	11.28
225k	6.97	6.32	6.55	10.51
300k	6.31	5.64	5.92	9.51
350k	5.68	5.07	5.32	8.55
400k	5.17	4.61	4.85	7.79
450k	4.73	4.24	4.49	7.16
500k	4.36	3.96	4.23	6.63
550k	4.05	3.71	3.96	6.18
600k	3.79	3.45	3.69	5.77
650k	3.55	3.24	3.47	5.41
675k	3.33	3.05	3.27	5.08
725k	3.15	2.84	3.03	4.77
750k	3.04	2.72	2.88	4.58

From this evaluation, it can be observed that the proposed model is 9.3% better throughput than VMI SCM [5], 12.8% better throughput than SC [10], and 12.3% better throughput than DDM SCM [13] for Jamming Attack. Similarly, average throughput for Sybil Attacks was evaluated, and can be observed from table 9 as follows:

Table 9. Average throughput for Sybil Attack w.r.t. number of blocks in the blockchain

NB	**T (bpm) VMI SCM [5]**	**T (bpm) SC [10]**	**T (bpm) DDM SCM [13]**	**T (bpm) SCMB QA**
24k	159.69	167.95	259.48	352.79
50k	101.05	110.92	157.03	217.17
75k	66.69	75.13	94.92	135.95
100k	41.23	46.32	58.21	83.65
125k	22.37	24.00	35.85	48.89
150k	14.00	14.00	23.47	31.33
176k	10.25	10.11	17.25	22.96
200k	8.85	9.13	15.05	19.96
225k	8.23	8.57	14.05	18.61
300k	7.45	7.67	12.70	16.84
350k	6.71	6.89	11.41	15.14
400k	6.11	6.27	10.41	13.80
450k	5.60	5.78	9.63	12.71
500k	5.16	5.42	9.05	11.80
550k	4.79	5.07	8.47	10.99
600k	4.48	4.73	7.90	10.28
650k	4.19	4.44	7.41	9.62
675k	3.94	4.19	6.98	9.03
725k	3.73	3.89	6.48	8.48
750k	3.60	3.71	6.17	8.13

From this evaluation, it can be observed that the proposed model is 31.5% better throughput than VMI SCM [5], 31.8%

better throughput than SC [10], and 8.5% better throughput than DDM SCM [13] for Sybil Attack. Similarly, evaluation of storage cost for mining is tabulated w.r.t. number of blocks used for mining (NB) in table 10, wherein Selfish Mining Attack model was used.

Table 10. Average storage cost needed by Selfish Mining Attack w.r.t. number of blocks in the blockchain

NB	S (MB) VMI SCM [5]	S (MB) SC [10]	S (MB) DDM SCM [13]	S (MB) SCMB QA
24k	3.3	3.7	2.2	1.1
50k	8.4	9.6	6.1	2.8
75k	22.0	25.2	17.3	7.6
100k	54.1	61.8	45.2	19.2
125k	164.3	187.8	123.9	55.2
150k	464.6	515.9	308.6	148.0
176k	1074.0	1170.0	670.1	333.6
200k	1562.7	1702.2	1010.0	494.6
225k	1704.0	1856.2	1135.8	548.3
300k	2060.3	2244.1	1347.9	656.5
350k	2575.9	2805.9	1682.3	819.9
400k	3148.4	3429.7	2054.3	1001.6
450k	3778.4	4115.5	2463.0	1201.5
500k	4389.2	4673.3	2829.2	1401.1
550k	5040.5	5252.6	3214.7	1614.8
600k	5817.1	6061.8	3707.9	1863.1
650k	6648.4	6928.9	4235.4	2128.6
675k	7535.3	7852.2	4798.4	2412.2
725k	8478.6	8836.0	5397.6	2713.7
750k	9506.2	9947.8	6061.3	3039.7

From this evaluation, it can be observed that the proposed model has 31.6% lower storage costthan VMI SCM [5], 34.4% lower storage costthan SC [10], and 20.6% lower storage costthan DDM SCM [13] for Selfish Mining Attack model. This is because of optimum miner selection, and incorporation of residual energy during selection of miner nodes. Similarly, average storage costneeded for Time Jacking Attack models was evaluated, and can be observed from table 11 as follows,

Table 11. Average storage costneeded Time Jacking Attack model w.r.t. number of blocks in the blockchain

NB	S (MB) VMI SCM [5]	S (MB) SC [10]	S (MB) DDM SCM [13]	S (MB) SCMB QA
24k	5.6	6.3	4.0	1.8
50k	14.4	16.5	11.0	4.9
75k	36.3	41.5	29.6	12.7
100k	101.9	116.5	79.8	34.9
125k	295.7	332.1	206.2	96.2
150k	737.9	809.8	472.1	231.6
176k	1307.4	1423.6	831.4	410.3
200k	1632.9	1778.2	1071.8	521.2
225k	1878.4	2045.8	1239.4	601.2
300k	2311.1	2516.6	1509.9	735.8
350k	2854.4	3109.1	1863.3	908.4
400k	3455.7	3764.2	2253.8	1099.2
450k	4078.2	4390.3	2643.5	1299.6
500k	4709.0	4958.7	3018.9	1506.0
550k	5422.6	5649.5	3456.0	1736.7
600k	6225.4	6486.9	3966.8	1993.6
650k	7085.4	7382.9	4512.7	2268.5
675k	7999.2	8335.0	5092.9	2560.4
725k	9138.0	9765.7	5885.6	2911.3
750k	10275.4	11052.5	6634.8	3268.8

From this evaluation, it can be observed that the proposed model has 46.8% lower storage cost than VMI SCM [5], 50.2% lower storage costthan SC [10], and 34.8% lower storage costthan DDM SCM [13] for Jamming Attack. Similarly, average storage costneeded for Sybil Attacks was evaluated, and can be observed from table 6 as follows,

Table 12. Average storage costneeded Sybil Attack w.r.t. number of blocks in the blockchain

NB	S (MB) VMI SCM [5]	S (MB) SC [10]	S (MB) DDM SCM [13]	S (MB) SCMB QA
24k	4.2	4.6	3.1	1.4
50k	10.6	12.0	8.7	3.8
75k	26.2	29.9	23.2	9.8
100k	74.3	84.4	62.6	27.1
125k	221.2	243.9	163.2	76.3
150k	562.6	601.1	376.7	186.6
176k	997.2	1057.1	663.1	330.6
200k	1236.2	1315.7	852.4	417.2
225k	1421.0	1513.5	985.5	480.8
300k	1751.8	1863.8	1202.0	589.8
350k	2164.4	2302.6	1482.8	728.1
400k	2621.2	2787.8	1793.6	881.2
450k	3096.6	3256.1	2104.7	1042.2
500k	3580.8	3686.1	2405.6	1208.4
550k	4125.4	4204.9	2755.9	1393.7
600k	4737.1	4828.0	3162.2	1599.8

650k	5391.8	5494.1	3597.2	1820.5
675k	6088.9	6204.0	4060.8	2055.5
725k	6945.9	7250.5	4689.4	2336.7
750k	7809.0	8205.6	5286.3	2623.4

From this evaluation, it can be observed that the proposed model has 43.5% lower storage cost than VMI SCM [5], 52.8% lower storage costthan SC [10], and 35.9% lower storage costthan DDM SCM [13] for Sybil Attack.Due to these improvements, the proposed model showcases high scalability, and better mining performance. This ensures that the model is applicable for high-speed, low energy, and high throughput sidechain-based SCM application deployments. As the evaluation was done for different types of attacks, it can be observed that the model's QoS performance is consistent, thus suggesting that it is capable of resisting these attacks. A performance reduction of less than 0.1% was observed under attacks, thereby suggesting that the proposed model is capable of reducing attack probability by over 99.95% under different attack types. This characteristic makes the model highly useful for a wide variety of SCM-deployment scenarios.

5. Conclusion

The proposed SCMBQA model uses a combination of machine learning with miner to entity relationship mapping for improved miner selection and sidechain creation process. This model is further embedded with a PoSC based consensus & incremental Q learning model, that assists in resisting against a wide variety of SCM based attacks. Due to use of these models, it is observed that the proposed method is 12.7% faster than VMI SCM [5], 15.4% faster than SC [10], and 18.2% faster than DDM SCM [13]under different attack scenarios. Due to incorporation of energy consumption during miner selection, the proposed model has 14.1% lower energy consumption than VMI SCM [5], 19.6% lower energy consumption than SC [10], and 16.8% lower energy consumption than DDM SCM [13] for different attack types. The model also incorporates throughput during selection of miner nodes, due to which, the proposed model is 14.1% better throughput than VMI SCM [5], 16.5% better throughput than SC [10], and 14.9% better throughput than DDM SCM [13] for under different types of attacks. Furthermore, the model incorporates sidechaining, due to which 31.6% lower storage cost than VMI SCM [5], 34.4% lower storage cost than SC [10], and 20.6% lower storage cost than DDM SCM [13]under different attacks is observed. Based on this performance enhancement, the proposed sidechaining model is useful for a wide variety of SCM deployments. In future, the model's performance can be validated on different attack types, which will assist in further scaling the model to multiple SCM types. Moreover, the model's QoS performance can be improved via use of deep learning-based miner selection methods, which can incorporate trust-levels and privacy levels for securing multiple types of SCM deployments.

References

[1] S. E. Chang and Y. Chen, "When Blockchain Meets Supply Chain: A Systematic Literature Review on Current Development and Potential Applications," in IEEE Access, vol. 8, pp. 62478-62494, 2020, doi: 10.1109/ACCESS.2020.2983601.

[2] R. W. Ahmad, K. Salah, R. Jayaraman, I. Yaqoob, M. Omar and S. Ellahham, "Blockchain-Based Forward Supply Chain and Waste Management for COVID-19 Medical Equipment and Supplies," in IEEE Access, vol. 9, pp. 44905-44927, 2021, doi: 10.1109/ACCESS.2021.3066503.

[3] S. Saberi, M. Kouhizadeh and J. Sarkis, "Blockchains and the Supply Chain: Findings from a Broad Study of Practitioners," in IEEE Engineering Management Review, vol. 47, no. 3, pp. 95-103, 1 thirdquarter,Sept. 2019, doi: 10.1109/EMR.2019.2928264.

[4] A. Shahid, A. Almogren, N. Javaid, F. A. Al-Zahrani, M. Zuair and M. Alam, "Blockchain-Based Agri-Food Supply Chain: A Complete Solution," in IEEE Access, vol. 8, pp. 69230-69243, 2020, doi: 10.1109/ACCESS.2020.2986257.

[5] I. A. Omar, R. Jayaraman, K. Salah, M. Debe and M. Omar, "Enhancing Vendor Managed Inventory Supply Chain Operations Using Blockchain Smart Contracts," in IEEE Access, vol. 8, pp. 182704-182719, 2020, doi: 10.1109/ACCESS.2020.3028031.

[6] K. Salah, N. Nizamuddin, R. Jayaraman and M. Omar, "Blockchain-Based Soybean Traceability in Agricultural Supply Chain," in IEEE Access, vol. 7, pp. 73295-73305, 2019, doi: 10.1109/ACCESS.2019.2918000.

[7] F. D. Valle and M. Oliver, "Blockchain Enablers for Supply Chains: How to Boost Implementation in Industry," in IEEE Access, vol. 8, pp. 209699-209716, 2020, doi: 10.1109/ACCESS.2020.3038463.

[8] B. Müßigmann, H. von der Gracht and E. Hartmann, "Blockchain Technology in Logistics and Supply Chain Management—A Bibliometric Literature Review From 2016 to January 2020," in IEEE Transactions on Engineering Management, vol. 67, no. 4, pp. 988-1007, Nov. 2020, doi: 10.1109/TEM.2020.2980733.

[9] D. Shakhbulatov, J. Medina, Z. Dong and R. Rojas-Cessa, "How Blockchain Enhances Supply Chain Management: A Survey," in IEEE Open Journal of the Computer Society, vol. 1, pp. 230-249, 2020, doi: 10.1109/OJCS.2020.3025313.

[10] I. A. Omar, R. Jayaraman, M. S. Debe, K. Salah, I. Yaqoob and M. Omar, "Automating Procurement Contracts in the Healthcare Supply Chain Using Blockchain Smart Contracts,"

in IEEE Access, vol. 9, pp. 37397-37409, 2021, doi: 10.1109/ACCESS.2021.3062471.

[11] A. Tharatipyakul and S. Pongnumkul, "User Interface of Blockchain-Based Agri-Food Traceability Applications: A Review," in IEEE Access, vol. 9, pp. 82909-82929, 2021, doi: 10.1109/ACCESS.2021.3085982.

[12] Q. Zhu and M. Kouhizadeh, "Blockchain Technology, Supply Chain Information, and Strategic Product Deletion Management," in IEEE Engineering Management Review, vol. 47, no. 1, pp. 36-44, Firstquarter,march 2019, doi: 10.1109/EMR.2019.2898178.

[13] W. Alkhader, K. Salah, A. Sleptchenko, R. Jayaraman, I. Yaqoob and M. Omar, "Blockchain-Based Decentralized Digital Manufacturing and Supply for COVID-19 Medical Devices and Supplies," in IEEE Access, vol. 9, pp. 137923-137940, 2021, doi: 10.1109/ACCESS.2021.3118085.

[14] Y. Fu and J. Zhu, "Big Production Enterprise Supply Chain Endogenous Risk Management Based on Blockchain," in IEEE Access, vol. 7, pp. 15310-15319, 2019, doi: 10.1109/ACCESS.2019.2895327.

[15] A. Musamih et al., "A Blockchain-Based Approach for Drug Traceability in Healthcare Supply Chain," in IEEE Access, vol. 9, pp. 9728-9743, 2021, doi: 10.1109/ACCESS.2021.3049920.

[16] P. Gonczol, P. Katsikouli, L. Herskind and N. Dragoni, "Blockchain Implementations and Use Cases for Supply Chains-A Survey," in IEEE Access, vol. 8, pp. 11856-11871, 2020, doi: 10.1109/ACCESS.2020.2964880.

[17] G. Subramanian, A. S. Thampy, N. V. Ugwuoke and B. Ramnani, "Crypto Pharmacy – Digital Medicine: A Mobile Application Integrated With Hybrid Blockchain to Tackle the Issues in Pharma Supply Chain," in IEEE Open Journal of the Computer Society, vol. 2, pp. 26-37, 2021, doi: 10.1109/OJCS.2021.3049330.

[18] M. N. M. Bhutta and M. Ahmad, "Secure Identification, Traceability and Real-Time Tracking of Agricultural Food Supply During Transportation Using Internet of Things," in IEEE Access, vol. 9, pp. 65660-65675, 2021, doi: 10.1109/ACCESS.2021.3076373.

[19] H. Chen, Z. Chen, F. Lin and P. Zhuang, "Effective Management for Blockchain-Based Agri-Food Supply Chains Using Deep Reinforcement Learning," in IEEE Access, vol. 9, pp. 36008-36018, 2021, doi: 10.1109/ACCESS.2021.3062410.

[20] X. Zhang et al., "Blockchain-Based Safety Management System for the Grain Supply Chain," in IEEE Access, vol. 8, pp. 36398-36410, 2020, doi: 10.1109/ACCESS.2020.2975415.

[21] F. M. Benčić, P. Skočir and I. P. Žarko, "DL-Tags: DLT and Smart Tags for Decentralized, Privacy-Preserving, and Verifiable Supply Chain Management," in IEEE Access, vol. 7, pp. 46198-46209, 2019, doi: 10.1109/ACCESS.2019.2909170.

[22] Kumar, A., Liu, R. and Shan, Z. (2020), Is Blockchain a Silver Bullet for Supply Chain Management? Technical Challenges and Research Opportunities. Decision Sciences, 51: 8-37. https://doi.org/10.1111/deci.12396

[23] Abidi, MH, Alkhalefah, H, Umer, U, Mohammed, MK. Blockchain-based secure information sharing for supply chain management: Optimization assisted data sanitization process. *Int J Intell Syst*. 2021; 36: 260-290. https://doi.org/10.1002/int.22299

[24] Yang, J, Ma, X, Crespo, RG, Martínez, OS. Blockchain for supply chain performance and logistics management. *Appl Stochastic Models Bus Ind*. 2021; 37: 429–441. https://doi.org/10.1002/asmb.2577

[25] Durach, C.F., Blesik, T., von Düring, M. and Bick, M. (2021), Blockchain Applications in Supply Chain Transactions. J Bus Logist, 42: 7-24. https://doi.org/10.1111/jbl.12238

[26] Zhang, E. Economic supply chain management of advanced manufacturing industry based on blockchain technology. *Security and Privacy*. 2021;e204. doi:10.1002/spy2.204

[27] Obeidat, R., Ispas, A., Aleodor, B., Bendic, V., Blockchain Technology—Applicability in the Traceability of a Product Throughout the Supply Chain. *Macromol. Symp.* 2021, 396, 2000270. https://doi.org/10.1002/masy.202000270

[28] Sternberg, H.S., Hofmann, E. and Roeck, D. (2021), The Struggle is Real: Insights from a Supply Chain Blockchain Case. J Bus Logist, 42: 71-87. https://doi.org/10.1111/jbl.12240

[29] Falcone, E.C., Steelman, Z.R. and Aloysius, J.A. (2021), Understanding Managers' Reactions to Blockchain Technologies in the Supply Chain: The Reliable and Unbiased Software Agent. J Bus Logist, 42: 25-45. https://doi.org/10.1111/jbl.12263

[30] Patelli, N. and Mandrioli, M. (2020), Blockchain technology and traceability in the agrifood industry. Journal of Food Science, 85: 3670-3678. https://doi.org/10.1111/1750-3841.15477

[31] Kamble, S.S., Gunasekaran, A., Subramanian, N. *et al.* Blockchain technology's impact on supply chain integration and sustainable supply chain performance: evidence from the automotive industry. *Ann Oper Res* (2021). https://doi.org/10.1007/s10479-021-04129-6

[32] Sharma, A., Bahl, S., Bagha, A.K. *et al.* Blockchain technology and its applications to combat COVID-19 pandemic. *Res. Biomed. Eng.* (2020). https://doi.org/10.1007/s42600-020-00106-3

[33] Lohmer J., Petzok L., Lasch R. (2021) Governance design of blockchain consortia for efficient and transparent procurement and supply chain management. In: Bode C., Bogaschewsky R., Eßig M., Lasch R., Stölzle W. (eds) Supply Management Research. Advanced Studies in Supply Management. Springer Gabler, Wiesbaden. https://doi.org/10.1007/978-3-658-35449-7_6

[34] Chowdhury, S., Rodriguez-Espindola, O., Dey, P. *et al.* Blockchain technology adoption for managing risks in operations and supply chain management: evidence from the UK. *Ann Oper Res* (2022). https://doi.org/10.1007/s10479-021-04487-1

[35] Nayal, K., Raut, R.D., Narkhede, B.E. *et al.* Antecedents for blockchain technology-enabled sustainable agriculture supply

chain. *Ann Oper Res* (2021). https://doi.org/10.1007/s10479-021-04423-3

[36] Alazab, M., Alhyari, S., Awajan, A. *et al.* Blockchain technology in supply chain management: an empirical study of the factors affecting user adoption/acceptance. *Cluster Comput* **24,** 83–101 (2021). https://doi.org/10.1007/s10586-020-03200-4