POLITECNICO DI TORINO
Repository ISTITUZIONALE

Blockchain-Based Transaction Management in Smart Logistics: A Sawtooth Framework

Original
Blockchain-Based Transaction Management in Smart Logistics: A Sawtooth Framework / Perboli, Guido; Capocasale, Vittorio; Gotta, Danilo. - ELETTRONICO. - (2020), pp. 1713-1718. (2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC)July, 13-17) [10.1109/COMPSAC48688.2020.000-8].

Availability:
This version is available at: 11583/2848263 since: 2020-10-14T16:44:40Z

Publisher:
ieee

Published
DOI:10.1109/COMPSAC48688.2020.000-8

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

08 May 2026

Blockchain-based transaction management in Smart Logistics: a Sawtooth framework

1st Guido Perboli
ICELab@Polito & CARS@Polito
Politecnico di Torino
Turin, Italy
https://orcid.org/0000-0001-6900-9917

2nd Vittorio Capocasale
ICELab@Polito & CARS@Polito
Politecnico di Torino
Turin, Italy
vittorio.capocasale@studenti.polito.it

3rd Danilo Gotta
Service Innovation
TIM
Turin, Italy
danilo.gotta@telecomitalia.it

***Abstract*—Blockchain is a disruptive technology that can be adopted in several business models. However, its applicability in the Supply Chain and in the context of the Logistics 4.0 and Smart Logistics revolution in particular, must still be proved from both an economic and an efficiency standpoint. This paper describes a Hyperledger Sawtooth-based framework for Supply Chain and Smart Logistics. The performance evaluation tests are performed on two Smart Logistics system settings. The results underline the performance decay of the system when concurrent transactions are submitted to multiple nodes.**

***Index Terms*—Blockchain, Smart Logistics, Logistics 4.0, Sawtooth, performance measure**

I. Introduction

Blockchain (BC) emerged as a leading technology layer for financial applications. Nevertheless, in the past years, the attention of researchers and practitioners moved to the application of the BC technologies to other domains [17, 18, 23]. In this context, Supply Chain (SC) and Logistics are the topics paying more attention to the BC, with the creation of several startups and the introduction of the BC in the agenda of countries and companies [1].

BC is a disruptive innovation, due to its capability of ensuring data immutability and public accessibility of data streams. Moreover, its decentralized and distributed infrastructure prevents the problems of the present centralized approaches, including trust issues, such as fraud, corruption, tampering and falsify information, and their limited resiliency. Centralized systems are vulnerable to collapse since a single point of breakdown might lead the whole system to be crashed. One of the more promising fields comes from the SC management, and the Logistics in particular. In fact, Logistics is considered by the different actors as the "reason to be" of each firm belonging to a SC. Without Logistics, no raw material can be extracted, transformed and delivered to the final user [9, 26]. Logistics is evolving rapidly in the past decade, thanks to the introduction of new management frameworks, as the Physical Internet and Industry 4.0, and new technologies, mainly ICT-based, as the Internet of Things (IoT), Business Analytics, Artificial Intelligence, and BC companies [9]. All these challenges have a common need: the benefits come from the network effect, and this can be obtained only with a proper sharing of the information. On the other side, the data sharing must be secured, distributed (e.g., for optimizing the subsystems locally) and with some auto-mated actions related to the different regulations and negotiations. Thus, BC appears as a natural technology for implementing these common issues. Presently, the main limits are the issues related to the scalability and the costs of the BC and the few use cases with clear costs/benefits analysis. Moreover, the literature mainly considers the Business Process Modelling and the Technology Design Process of a Block-chain-based solution the reference frameworks lack in terms of standard methodology to design, develop and validate the overall BC solution at the Strategic level. The interested reader can refer to [27, 23] for a more complete review of these issues.

However, since this technology is still in its early stages, it presents some inherent defects and its deployment in factual SC and Logistics applications is somehow problematic. In particular, there is a general lack in the literature of BC. Despite its importance, just a few papers deal with non-finance implementations [23], with a specific focus in the food industry, due to regulations and traceability issues [19, 22, 27]. More in general, the literature mainly considers the Business Process Modelling and the Technology Design Process of a Block-chain-based solution [5, 6, 10, 24], while the main reference for the design, development and validation the overall BC solution at the Strategic level is the framework presented in [27]. Aim of this paper is to present a solution for Smart Logistics based on the Sawtooth BC, a verticalization of the Hyperledger technology specifically developed for managing logistic applications and to integrate them with IoT devices. In particular, we will show the results of the tests on different logistic configurations, discussing advantages and issues of the solution.

II. Logistics 4.0 and Blockchain

In the current economy *"competition is no longer between organizations, but among supply chains"* [20]. For this reason, it is crucial to improve the SC integration [31], defined as:

> *"the alignment, linkage and coordination of people, processes, information, knowledge, and strategies across the supply chain between all points of contact and influence to facilitate the efficient and effective*

flows of material, money, information, and knowledge in response to customer needs" [16].

In Logistics, the core of the SC [27], customers are requiring more customized products and lower delivery times, while warehouses and carriers are often times half-filled and many products never reach the market because *"supply chains are usually plagued with lack of both timely and correct information"* [31].

Fortunately, in the recent years the evolution of technology has brought what is defined as the fourth revolution in the industrial world, which is leading to the adoption of the term *"Logistics 4.0"* to describe the application of these new technologies in Logistics [12, 29, 31, 35]. The main aspects of this revolution are: the decreased size, imprecision and expensiveness of sensors which allows to gather an enormous quantity of data and to identify and locate every entity in the system; the development of data mining and machine learning techniques, which help to extract relevant information from the gathered data; the Internet of Things (IoT) and Internet of Services (IoS), which enable the communication of smart devices and allow them to autonomously take decisions according to the relevant information extracted. The combination of these technologies creates a completely automated and self-adapting system (cyber-physical system) [12, 15, 31, 35].

In other words, Logistics 4.0 is a revolution based on information sharing and digitalization [29, 35]. For this reason, while this revolution is particularly relevant in the context of a single organization [29, 33, 35], its extension to the whole Supply Chain introduces additional challenges. In fact, *"there is a shift from traditional supply chains to open supply chain network, from long-lasting business relationships to short-term business connections"* [12] (Fig. 1), and sharing information in such a context requires additional guarantees about the data security, accessibility, reliability and distribution [27].

Among the available technologies, the Distributed Ledger (DL) ones are particularly suited to solve the problem, because they allow managing a distributed, append-only registry in a shared way. Of all the DL technologies, this paper focuses on the BC one, which provides the following properties:

- **autonomy:** because each entity (node) can submit transactions without relying on trusted third parties [2, 21];
- **authenticity:** because transactions are digitally signed [14, 27, 36];
- **immutability:** as a consequence of the hash mechanism used to link blocks [2, 21, 37];
- **transparency and auditability:** because the ledger stores the history of the data and not just its current value. Moreover, each node has a private copy of the ledger, so it has direct access to the data [2, 21, 36, 37];
- **redundancy and persistency:** because the ledger is replicated in many nodes [36, 37].
- **resiliency:** because to corrupt the ledger it would be necessary to coherently modify the majority of its copies [14, 21].

As a consequence, BC would enforce the standardization of a common interface to exchange authentic, transparent and immutable data among non-trusting parties without the involvement of any third one [27, 23]. In addition to the end-to-end automation, the BC introduction may reduce paperwork processing time and costs [14, 18, 34], reduce the number of counterfeits [14, 34] and simplify the tracking of products, which would make easier to track down contamination sources or production defects [14]. Moreover, it would discourage unethical or unsustainable actors' behavior [28].

At the same time, both [28, 34] well identify the main barriers that may negatively affect the adoption of the BC technology in the SC and Logistics context, which may be: intra-organizational, like the reluctance in replacing working *"legacy"* systems and the costs of the migration toward the BC solution; inter-organizational, like the concerns about the information disclosure to non-trusting parties; system-related, like BC's immaturity and its related security flows; external, like the lack of governmental regulations and the market competition and uncertainty, which may not reward the investments in the BC technology.

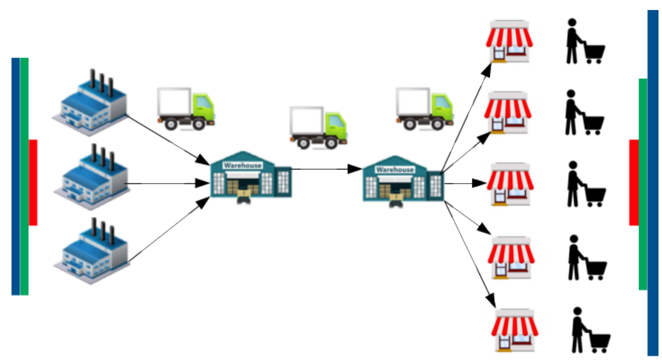

Fig. 1. While modern logistic chains are more similar to networks than to sequential links, this paper considers a simplified version of them. In particular, this figure shows the three use cases used for the testing of the Full System: a single producer and a single consumer (red, use case 1); three producers and three consumers (green, use case 2); three producers and five consumers (blue, use case 3).

III. Blockchain platforms

Many BC platforms support the definition of smart contracts[1] and thus the implementation of a Smart Logistics system:

- **Hyperledger Fabric** is neither fully decentralized nor completely BFT[2] [30];
- **Quorum** offers only two consensus algorithms: Raft and IBFT [7]. Raft is CFT only while IBFT has relevant scalability limitations [11];

[1]A smart contarct is a tamper-proof program.

[2]In a distributed system, for consistency reasons, the nodes must reach an agreement on the state of the system itself. A consensus algorithm is what allows them to reach such an agreement, and it can be either CFT (resilient to the crash of some nodes) or BFT (resilient to the crash or malicious behavior of some nodes).

- **Corda** is mostly renowned in the financial sector [4], and its peculiar implementation does not fit well in the context of a single supply chain [13];
- **Hyperledger Sawtooth** integrates a consensus algorithm that targets large distributed node populations with minimal resource consumption, namely, the Proof of Elapsed Time (PoET) and it allows the submission of groups of transactions that must be executed as a whole (batches). Moreover, specific templates and rules for Supply Chain and Logistics are present.

In real Logistics and SC solutions the maturity level of the code, as well as the possibility to have commercial support are essential. Moreover, the presence of templates specifically designed for Logistics and Supply Chain and the presence of APIs for integrating IoT devices is another requirement. The aforementioned issues restrict the choice to Hyperledger Fabric and Sawtooth.

In our solution, we decided to adopt Hyperledger Sawtooth for its PoET algorithm, which is suitable for large-scaled Supply Chains and the templates for Supply Chain management, even if Sawtooth is the only of the four that does not allow to build private DLs between specific partners over the main one [25]. Nonetheless, this platform is the best suited for the purpose: overcoming the aforementioned limitation is possible by integrating a system as the one described in [3], specifically designed for sharing private data over public DLs by cryptographic APIs.

IV. Smart Logistics system settings

In our application, we considered two different system settings: a very basic one, named Simple System, and a more realistic one, named Full System.

A. Simple System

The Simple System defines a minimal set of entities and operations and it has to be consider as an essential (but efficient) implementation of an exchange-based framework:

- **actor:** it is any entity that submits transactions to the system. An actor could be a buyer, a seller, a company, an IoT device and more;
- **asset:** it represents the general concept of exchangeable property: it has a state and an owner;
- **proposal:** it is the smart contract that regulates the exchanges of assets. The exchange takes place only when both the buyer and the seller agree on it. In some cases, this operation could be performed automatically by IoT devices, for example in the case of shipments of containers, as described in [8].

B. Full System

The Full System extends the simple one by defining new entities and more complex operations:

- **actor:** it is any entity that submits transactions to the system. An actor could be a buyer, a seller, a company, an IoT device and more;
- **asset:** it represents the general concept of exchangeable property, and it is described by many attributes that it is possible to update: position, temperature, volume, owner and keeper (the actor which has custody of the asset). A tamper-proof IoT device could be in charge to update the position or the temperature of the asset.
- **delivery:** it represents groups of assets that must be moved together. It has a state and it tracks the assets being moved and their position, as well as information about its sender, its keeper and its receiver;
- **carrier:** it is the entity that performs the deliveries, and it divides them between the ones that are completed and the ones that are not. It is described by its position and the quantity of fuel consumed and of CO_2 emitted. These values could be updated by a tamper-proof IoT device;
- **warehouse:** it represents a facility where the assets can be stored. It is characterized by its available volume, the number of carriers ready to be loaded and the number of free docking spots. All these values could be updated by IoT sensors;
- **policy:** except for the actors, all the entities in the system are regulated by policy. A policy defines which actors can perform which operations on which entities. It allows to implement the concepts of shared property, custody and operation delegation;
- **proposal:** it is the smart contract that regulates the modifications of the state of the ledger performed by groups of actors. Differently from IV-A, a smart contract may be necessary not only in the case of exchanges of assets but also for simple updates of the properties of an entity, according to the policy that regulates it.

V. Test of the Sawtooth-based Blockchain architecture

This section describes the tests performed on the two implemented Smart Logistics systems. According to [32], first the general system description is given, then the tests performed on the Simple System are described, and finally the ones performed on the Full System in three different use cases.

A. Test environment

The tests are performed using a single computer and the Docker platform to virtualize a BC network of five validators[3]. The transaction processors are implemented in Go, while the clients in Typescript.

1) Hardware configuration:

- MODEL: ASUS N56JK-CN051H;
- CPU: Intel Core i7-4710HQ, 2.50GHz, octacore;
- RAM: 7,7 GiB DDR3, 1600 MT/s;
- DISK: 343,0 GB, 5400 RPM.

[3]In the context of the Sawtooth framework, a *validator* is a node which participates to the consensus process. A node that can only submit transactions is a *client*.

2) Software configuration:

- Ubuntu: 18.04.3 LTS;
- Docker: version 19.03.3, build a872fc2f86;
- Sawtooth: 1.0.5;
- Go: version go1.11.2 linux/amd64;
- Node: v11.0.0;
- Angular: 8.3.1;
- Google Chrome: 70.0.3538.77, launched with the "–disable-web-security" command line option to avoid problems related to the CORS policy.

3) Network configuration:

- Consensus protocol: PoET-CFT;
- Geographic distribution: co-located nodes;
- Network model: 5-node complete graph;
- Number of nodes involved in the test transaction: 5;
- Software component dependencies: none, other than the default ones.

4) Blockchain properties configuration:

- sawtooth.poet.target_wait_time: 5;
- sawtooth.poet.initial_wait_time: 25;
- sawtooth.publisher.max_batches_per_block: 1000;
- sawtooth.validator.max_transactions_per_block: 1000;
- sawtooth.poet.ztest_minimum_win_count: 999999999.

5) Validators' properties configuration:

- peering: dynamic;
- scheduler: parallel;
- network: trust;

B. Methodology

1) Test tools and frameworks: the tests are performed in a local environment, thus the client is hosted on the same machine of the network of validators. Network load is generated and captured using the Angular framework and the Google Chrome web browser. The node used by the client to submit the transactions changes in each test. This information is thus provided with the description of each of the tests performed.

2) Workload: for the Simple System the workload is a combination of almost CRUD operations. For the Full System, the workload is composed of flows of assets from a producer to a customer.

3) Finality threshold: the finality threshold is 100% of the nodes: all the validators must consider a transaction committed before it is considered as such by the client.

4) Measure type: the focus of this work is on the transaction throughput measure (TPS), defined as: *total committed transactions / total time in seconds* [32].

5) Observation points: the BC performance is measured from the perspective of a client. The total time to calculate the TPS measure is thus defined as: *time the client reads the transaction as committed on all the nodes - time the client submits the transaction to one node*.

6) Testing strategy: the tests are performed using a polling strategy: the reads performed to poll the system help the simulation because in a real-case scenario both transactions and reads should be submitted by the clients. During the tests, the network traffic quantity is kept almost constant, by allowing only up no N batches to be pending. Various values of N are used in the tests. All the batches used in each test are prepared in advance so that they can be submitted immediately without spending computational time on the client. Due to the memory constraints of the computer used, this approach limits the batch sizes used in some tests, as showed in Fig. 2.

7) Transactions characteristics: the transactions used for testing purposes can all be considered small and simple: even transactions that are linear in the number of entities defined in the system can be considered simple as a consequence of the limited amount of such entities. The dependencies and data access patterns of the transactions follow the ones of a simple production use.

C. Simple System

Three types of tests were performed on the Simple System:

- single batch tests: only one batch is submitted to the system (N=1), and thus all the transactions are submitted to the same validator. The results are reported in Fig. 2;
- batch dependency tests: the client submits each batch to a different validator, in a round-robin fashion. Twenty batches are submitted in total, each containing a different number of transactions in each test. The client allows up to three batches to be not yet committed (N=3). Each batch is dependent on (can be committed only after) the previous one, so a total order on the batches is enforced. The results are shown in Fig. 2.
- batch concurrency tests: as for the previous type, but with the differences that the client allows up to ten batches to be not yet committed (N=10) and that the batches are independent of one another. The results are shown in Fig. 2.

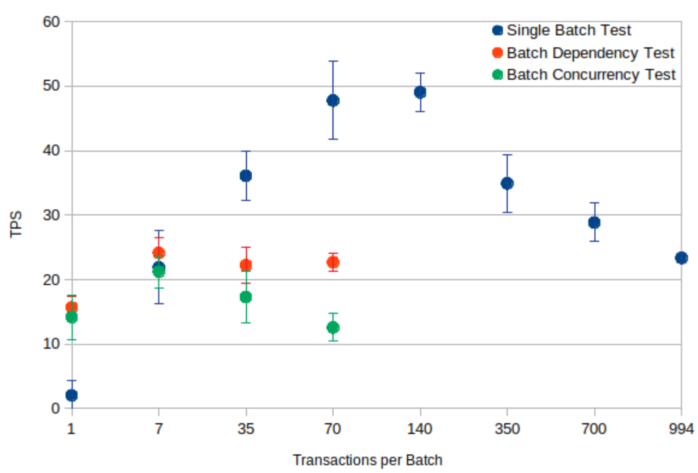

Fig. 2. Simple System: transactions per second (TPS) in the tests performed.

D. Full System

The Full System was tested in three use cases:

- the first use case presents a single producer, a single retail company with two warehouses and a single customer

(Fig. 1). A total order is enforced among the batches. All the batches are submitted to the same validator;

- the second use case presents three producers, a single retail company with two warehouses and three customers (Fig. 1). This use case is characterized by a greater number of parallel deliveries with respect to the previous one, because the assets can be produced by different sources and can reach different destinations. The batches are submitted to three different validators;
- the third use case presents three producers and five customers (Fig. 1). As a consequence of the bigger number of customers with respect to the previous use case, the number of concurrent deliveries is also increased. The batches are submitted to all the validators.

For each use case, a total of twenty batches is submitted to the system, and each batch contains thirty-five transactions. The client also counts the total number of reads it performs. Each test includes a total of fifty repetitions. The results of the tests are reported in Fig. 3. The read per second (RPS) value represents the workload the system was subject to during the tests, as a consequence of the polling strategy, and not the maximum number of reads the system is able to process.

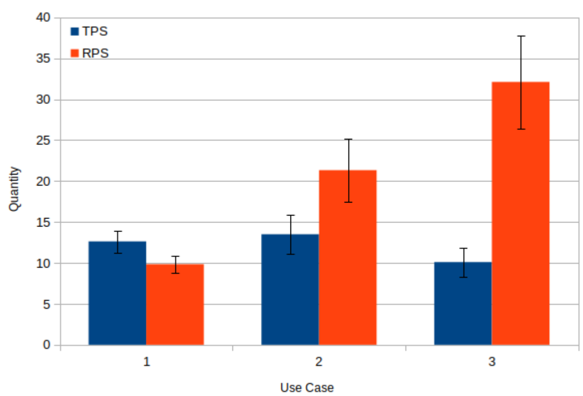

Fig. 3. Full System: transactions per second (TPS) and reads per seconds (RPS) in the three use cases for a network of five nodes.

VI. Results highlights and conclusion

This paper, after a brief discussion about the potential adoption of the Blockchain technology as a tool to propagate the Logistics 4.0 revolution to the whole supply chain, provides some data about the performance of two Smart Logistics systems implemented through the Sawtooth framework.

While the TPS values obtained are strictly bound to the specific hardware, software and network configuration used, some more general considerations emerge from a comparative analysis of the results obtained in the various tests and between the two systems.

In both the systems the TPS value decreases when the number of independent transactions submitted concurrently to different validators is too high. This behavior is probably characteristic of the PoET consensus algorithm, and in particular it is probably a consequence of the *”sawtooth.poet.target_wait_time”* value. This setting expresses the likelihood for a node to publish a block: a high value decreases the maximum reachable TPS while a small one increments the risk of the creation of forks. For this reason, it is probably possible to improve the performance of the systems in the batch concurrency tests (Fig. 2) and in the third use case (Fig. 3) by increasing the value of this setting.

Fig. 2 shows how the number of transactions in each batch impacts the TPS value. The low performance of small batches is probably linked to the batch encapsulation overhead and to the low system load, while for the bigger ones it is probably linked to the increased required processing time which slows down the block propagation across the network, which makes forks more likely. All these assumptions, however, should be verified by further studies and tests.

Future researches could be aimed at overcoming some limitations of the systems: a better management of the reading and writing sets of the transactions could improve their concurrency and a more use case specific design could allow to trade some flexibility for performance.

References

[1] Alice Consortium, “Alice recommendations to h2020 work programs 2018-2020,” 2016.

[2] A. Baliga, “The blockchain landscape,” *Persistent Systems*, 2016.

[3] F. L. Bellifemine, D. Gotta, and T. Trucco, “System and method for managing supply of electric energy through certified measures,” WO Patent WO2 019 081 298A1, may 2, 2019.

[4] R. G. Brown, “The corda platform: An introduction,” *Retrieved*, vol. 27, p. 2018, 2018.

[5] M. Caro, M. Ali, M. Vecchio, and R. Giaffreda, “Blockchain-based traceability in agri-food supply chain management: A practical implementation,” in *2018 IoT Vertical and Topical Summit on Agriculture-Tuscany (IOT Tuscany), Tuscany, Italy*, vol. 319, 2018, pp. 1–4.

[6] R. Casado-Vara, A. Gonzalez-Briones, J. Prieto, and J. Corchado, “Smart contract for monitoring and control of logistics activities: Pharmaceutical utilities case study,” *Advances in Intelligent Systems and Computing*, vol. 771, pp. 509–517, 2019.

[7] chris-j-h. Quorum whitepaper. [Online]. Available: https://github.com/jpmorganchase/quorum/blob/master/docs/Quorum%20Whitepaper%20v0.2.pdf

[8] K. Christidis and M. Devetsikiotis, “Blockchains and smart contracts for the internet of things,” *IEEE Access*, vol. 4, pp. 1–1, 01 2016.

[9] T. G. Crainic, G. Perboli, and M. Rosano, “Simulation of intermodal freight transportation systems: a taxonomy,” *European Journal of Operational Research*, vol. 270, no. 2, pp. 401–418, 2018.

[10] C. Di Ciccio, A. Cecconi, J. Mendling, D. Felix, D. Haas, D. Lilek, F. Riel, A. Rumpl, and P. Uhlig, “Blockchain-based traceability of inter-organisational business pro-

cesses," *Lecture Notes in Business Information Processing*, pp. 1–4, 2018.
[11] N. Diarra, "Choosing a consensus protocol for uses cases in distributed ledger technologies," in *2019 Sixth International Conference on Software Defined Systems (SDS)*. IEEE, 2019, pp. 306–309.
[12] L. Domingo Galindo, "The challenges of logistics 4.0 for the supply chain management and the information technology," Master's thesis, NTNU, 2016.
[13] G. Greenspan. R3 corda: Deep dive and technical review. [Online]. Available: https://www.multichain.com/blog/2018/05/r3-corda-deep-dive-and-technical-review/
[14] N. Hackius and M. Petersen, "Blockchain in logistics and supply chain: trick or treat?" in *Digitalization in Supply Chain Management and Logistics: Smart and Digital Solutions for an Industry 4.0 Environment*, 2017, pp. 3–18.
[15] E. Hofmann and M. Rüsch, "Industry 4.0 and the current status as well as future prospects on logistics," *Computers in industry*, vol. 89, pp. 23–34, 2017.
[16] M. Johnson and G. Stevens, "Integrating the supply chain... 25 years on," *International Journal of Physical Distribution & Logistics Management*, vol. 46, 02 2016.
[17] H. Kakavand, N. Sevres, and B. Chilton, "The blockchain revolution: An analysis of regulation and technology related to distributed ledger technologies," *SSRN Electronic Journal*, 01 2017.
[18] K. Korpela, J. Hallikas, and T. Dahlberg, "Digital supply chain transformation toward blockchain integration," in *proceedings of the 50th Hawaii international conference on system sciences*, 2017.
[19] K. Leng, Y. Bi, L. Jing, H.-C. Fu, and I. Van Nieuwenhuyse, "Research on agricultural supply chain system with double chain architecture based on blockchain technology," *Future Generation Computer Systems*, vol. 86, pp. 641–649, 2018.
[20] S. Li, B. Ragu-Nathan, T. Ragu-Nathan, and S. S. Rao, "The impact of supply chain management practices on competitive advantage and organizational performance," *Omega*, vol. 34, no. 2, pp. 107–124, 2006.
[21] I.-C. Lin and T.-C. Liao, "A survey of blockchain security issues and challenges," *International Journal of Network Security*, vol. 19, pp. 653–659, 09 2017.
[22] D. Mao, F. Wang, Z. Hao, and H. Li, "Credit evaluation system based on blockchain for multiple stakeholders in the food supply chain," *International journal of environmental research and public health*, vol. 15, no. 8, p. 1627, 2018.
[23] S. Musso, G. Perboli, M. Rosano, and A. Manfredi, "A decentralized marketplace for m2m economy for smart cities," in *2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*. IEEE, 2019, pp. 27–30.
[24] M. Nakasumi, "Information sharing for supply chain management based on block chain technology," in *2017 IEEE 19th Conference on Business Informatics (CBI)*, vol. 1. IEEE, 2017, pp. 140–149.
[25] K. Olson, M. Bowman, J. Mitchell, S. Amundson, D. Middleton, and C. Montgomery, "Sawtooth: An introduction," 2018.
[26] P. Pawlewski and A. Greenwood, *Process Simulation and Optimization in Sustainable Logistics and Manufacturing*. Springer, 2014.
[27] G. Perboli, S. Musso, and M. Rosano, "Blockchain in logistics and supply chain: a lean approach for designing real-world use cases," *IEEE Access*, vol. PP, pp. 1–1, 10 2018.
[28] S. Saberi, M. Kouhizadeh, J. Sarkis, and L. Shen, "Blockchain technology and its relationships to sustainable supply chain management," *International Journal of Production Research*, vol. 57, no. 7, pp. 2117–2135, 2019.
[29] N. Schmidtke, F. Behrendt, L. Thater, and S. Meixner, "Technical potentials and challenges within internal logistics 4.0," in *2018 4th International Conference on Logistics Operations Management (GOL)*. IEEE, 2018, pp. 1–10.
[30] P. Sitoh. What are the differences between ethereum, hyperledger fabric and hyperledger sawtooth? [Online]. Available: https://medium.com/coinmonks/what-are-the-differences-between-ethereum-hyperledger-fabric-and-hyperledger-sawtooth-5d0fc279d862
[31] J. O. Strandhagen, L. R. Vallandingham, G. Fragapane, J. W. Strandhagen, A. B. H. Stangeland, and N. Sharma, "Logistics 4.0 and emerging sustainable business models," *Advances in Manufacturing*, vol. 5, no. 4, pp. 359–369, 2017.
[32] The Hyperledger Performance and Scale Working Group. Hyperledger blockchain performance metrics. [Online]. Available: https://www.hyperledger.org/resources/publications/blockchain-performance-metrics
[33] E. Tijan, S. Aksentijević, K. Ivanić, and M. Jardas, "Blockchain technology implementation in logistics," *Sustainability*, vol. 11, no. 4, p. 1185, 2019.
[34] Y. Wang, J. H. Han, and P. Beynon-Davies, "Understanding blockchain technology for future supply chains: a systematic literature review and research agenda," *Supply Chain Management: An International Journal*, 2019.
[35] S. Winkelhaus and E. H. Grosse, "Logistics 4.0: a systematic review towards a new logistics system," *International Journal of Production Research*, vol. 58, no. 1, pp. 18–43, 2020.
[36] K. Wüst and A. Gervais, "Do you need a blockchain?" in *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*. IEEE, 2018, pp. 45–54.
[37] Z. Zheng, S. Xie, H.-N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: A survey," *International Journal of Web and Grid Services*, vol. 14, p. 352, 10 2018.