Blockchain Implementation for Analysis of Carbon Footprint across Food Supply Chain

Denisolt Shakhbulatov†, Arshit Arora,† Ziqian Dong*, and Roberto Rojas-Cessa‡
*Department of Electrical and Computer Engineering
†Department of Computer Science
College of Engineering and Computing Sciences
New York Institute of Technology, New York, NY 10023
‡Department of Electrical and Computer Engineering
New Jersey Institute of Technology, Newark, NJ 07102

Email: {dshakhbu, aarora08, ziqian.dong}@nyit.edu, rojas@njit.edu

***Abstract*—The growing global population and the consequential increasing food demand have had a great impact on the environment and thus on the climate. Efforts are being made to reduce carbon footprint to mitigate such effects. Calculating the carbon footprint of food products is complex and requires the cooperation of all the stakeholders of the food supply chain. A record keeping system for tracking carbon footprint while preserving privacy for the related parties is needed. This paper presents a new implementation of blockchain for tracking of carbon footprint on food production and transportation stages. We designed a system that tracks the carbon footprint of food processing facilities and transportation parties using cluster-based record keeping while preserving their privacy. We implemented the proposed carbon footprint chain and evaluated its throughput and latency under different scenarios. We show that our blockchain implementation is capable of operating with a larger number of nodes without any scalability issues.**

***Index Terms*—Food production, Carbon footprint, Life Cycle Assessment, Supply chain, Freight movement, Blockchain, RAFT, Consensus Algorithm.**

I. Introduction

The world population is estimated at 7.7 billion in 2019 and is projected to reach 9.7 billion by 2050 [1]. Recent studies have shown that not only the Earth's population is increasing, but also the per-capita food demand [2], [3]. On a global average, an adult in 2014 was 14 percent heavier, 1.3 percent taller, and had a 6.1 percent higher calorie demand than the average adult in 1975 [3]. The increasing demand for food has resulted in an industrialized food production system around the globe that has significantly impacted the land use (deforestation for agricultural farms), the climate (carbon emissions from agricultural activities), and the environment (agricultural runoffs) for a sustainable growth [2].

Efficient management of the food supply chain has the potential to reduce its carbon footprint (CF), environmental impact, and ensure food safety [4]. Many research studies have attempted to account for the carbon footprint of food supply chains [5], [6]. However, the interconnection of the stages of supply chain involves multiple stakeholders from production, processing, transportation, distribution, consumption and waste disposal with often conflicting interests, making it hard to retrieve fine-grained data for an accurate analysis [7].

Researchers often use life-cycle assessment to evaluate the environmental impact from all the stages of a product's life cycle [8], [9]. For food products, a significant amount of carbon is generated by agricultural farming activities [10]. Processing, storage, packaging, and distribution of food also contribute to its carbon footprint. Among all the life cycle stages, transportation of food plays a critical role in connecting all of its life cycle stages. It is reported that transportation of food accounts for 11% of all greenhouse gas emissions during the food life cycle in the United States [11]. However, lack of access to data in transportation reduces the accuracy of the estimation of the carbon footprint of this stage [12].

The Freight Analysis Framework, provided by the U.S. Department of Transportation, integrates data from agriculture, extraction, utility, construction, service, and other sectors to provide a view of freight movements among states and major metropolitan areas by all modes of transportation [13]. This data has been used for estimating freight movements across different states and regions. However, the available data does not provide fine-grained product categories for accurate calculation of the generated carbon footprint for each product. Furthermore, the actual distribution routes the freight moves from the source to the destination are unknown. Therefore, there is a need for the availability of the transportation data for an accurate calculation of the total amount of carbon footprint from a single product across its life cycle.

To address the issue of tracking carbon footprint for the transportation of food, we propose a new blockchain framework, called carbon footprint chain (CFC), to provide information on the pre-consumption stages of the food life cycle (from farm to processing, production, packaging, distribution and retail stages). The proposed CFC uses private clusters for verification of occurrence of events in each of the stages of the food products' life cycle. This cluster ensures that there is no single large blockchain, which is difficult to scale. The cluster-based approach also makes it hard to corrupt the blockchain due to the constant change of leaders of each cluster, and the fact that corrupted node can only corrupt a single cluster, but not the whole blockchain. Our approach uses a Raft-like blockchain consensus algorithm to achieve

978-1-7281-4693-5/19/$31.00 ©2019 IEEE
DOI 10.1109/Blockchain.2019.00079

high performance and scalability. Blockchain provides a means for stakeholders from the food supply chain to publish their data on transportation without disclosing private information. While taking into account the security vulnerabilities and a large number of facilities for each of the life cycle stages, the CFC is built on a decentralized network to ensure it's scalable and secure.

The remainder of the paper is organized as follows. Section II reviews the related work and common issues in blockchain implementations. Section III presents the design details of the proposed system including network structure, validation protocol, and data structure. Section IV presents the experimental set up used in the evaluation of the CFC and the obtained results. Section V concludes the paper.

II. Related Work

A. Life Cycle Assessment

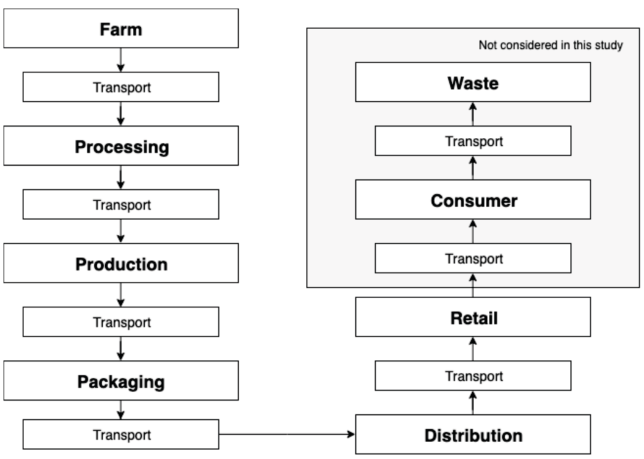

Fig. 1: Life cycle stages.

Life cycle assessment (LCA) is a tool that evaluates the carbon footprint of a product throughout its life cycle [14]. The stages considered in a general product life cycle are raw material extraction, production, processing, transportation, storage, consumption, waste disposal, and recycling. The boundaries of LCA are defined at the time of the study. Often, data from the food transportation stage is not accounted for in LCA. In some studies, rough estimates based on international or statewide trade data are used instead. However, transportation is not only a life cycle stage, but a common intermediate phase as food transfers from one stage to the next stage of the produce life cycle. In our study, we focus on examining the carbon footprint generated by the transportation of food.

B. Blockchain

The proposed CFC provides a framework to record and share data among various stakeholders while maintaining their privacy. Blockchain is a distributed ledger that contains blocks of transaction records. Each participating peer, or node, holds and processes a copy of a block. A variety of data can be stored in a block. Each block is directly connected to a previous block, creating a chain. The Blockchain data are resistant to modification except when allowed by the majority of peers. Blockchain is based on the interaction principles of peer-to-peer networks and utilizes distributed hash tables. If a node modifies the data stored on its local blockchain, it is detected during the validation process as the hash of the previous block will not match any other node's blockchain. This validation process forces the node to either request the full blockchain from the cluster leader or leave the network. Here, we consider each blockchain is processed by a cluster of nodes, where one node is the leader. Data modification can be performed only when over 51% of the nodes experience the same modification [15].

An application of blockchain in the food industry has focused on food traceability to provide a better tracking system to ensure food safety [16]. A study implemented a radio-frequency identification (RFID) and blockchain to track food products on all stages of its life cycle [17]. The authors used BigchainDB to resolve the scalability issues of traditional blockchains. Such implementations have already been in use by large corporations. IBM has also developed a blockchain tool for food traceability [16]. This tool uses a Hyperledger Fabric, which is another typical blockchain implementation, although not completely distributed [18]. None of the blockchain implementations in food industry are concerned with the calculation of the carbon footprint of food systems, which we address in this paper.

C. Common Issues in Traditional Blockchain Implementation

Conventional or Bitcoin blockchain faces a few significant issues in terms of scalability. Bitcoin blockchain processes only one transaction per second (tps), with a theoretical limit of 7 tps. Taking that into account, Bitcoin has a lower throughput than the majority of other networks used nowadays [19]. Bitcoin's block size is around 1MB with the total size of the entire network reaching over 200 Gigabytes in March 2019 [20]. In the case of supporting more transactions, similar to those in payment systems, the size of each block is expected to increase rapidly. This growth of blockchain is nonscalable [21]. Therefore, only large nodes with enough storage space and computation resources can participate in the consensus algorithm, centralizing the entire network, and leading to the opening of a security vulnerability. In this case nodes with large storage can dictate the decisions through a majority vote. Validation time for Bitcoin blockchain takes 10 minutes, which is longer than any other network [22]. The disadvantage of Hyperledger fabric is that it is a private blockchain, therefore it doesn't allow complete transparency. This blockchain implementation does not prevent corrupted data from being written into the Blockchain [23]. A novel blockchain implementation, Tendermint is commonly used in internet of things (IoT) blockchain implementations, however it represents a loose-First In First Out (FIFO) capability, resulting in out-of-order transactions [24]. The lack of ordered blocks would not work for our purpose of tracking the movement of food products from one life cycle stage to another because trucks transport food from facilities in one stage to the next in order. For example, a truck needs to first go through the farming facility to pick up goods to transport them to processing facilities. If the record has not been stored in the blockchain yet, then the truck will not be validated.

III. Proposed Carbon Footprint Chain Setup

In this section, we introduce our proposed carbon footprint chain (CFC) for tracking carbon footprint in the food systems. The proposed system is a lightweight and scalable Blockchain implementation that uses private clusters. We use CFC for recording and verifying the transactions of the life-cycle stages using a Raft-like consensus algorithm.

A. Life Cycle Assessment

Figure 1 shows the life cycle stages considered in this study. Our study primarily focuses on the transportation systems used in each of the food life cycles as they may provide more accuracy carbon footprint assessment. For simplicity, the consumption, waste disposal, and recycling stages are not included in this study but they will be considered in the life cycle assessment in future work.

B. Network

The CFC consists of private clusters, each representing a stage of the food life cycle. Nodes on each private cluster represent the facilities used on that stage. Because about 70% of food in the U.S. is transported by trucks, we represent transportation as the use of trucks as the sole sole form of transportation [25]. Nevertheless, the CFC system can be easily adapted to heterogeneous transportation means. We consider that trucks record the carbon footprint through a heavy-duty on-board diagnostics system. The recorded data such as type of goods, amount, mileage travelled, are sent to a cluster when a truck enters a new life cycle stage (e.g., when food arrives from a farm to a processing plant, or from a processing plant to a distribution center). Each life cycle stage has its own private blockchain cluster.

C. Block Formation

Each truck has an Electronic Logging Device (ELD) that collects data directly from the truck through a Heavy Duty On-Board Diagnostics (HD-OBD) device. According to the 2018 revisions by The California Air Resources Board (CARB), heavy-duty trucks are required to store their CO2 emission data while in operation. Storage of similar data for greenhouse gas emissions is already required on light-duty and medium-duty trucks [26].

The electronic logging device consists of a Raspberry Pi connected to the HD-ODB with an LTE Module for Internet connection and a GPS module for recording the actual route followed by the truck. The data collected on the ELD consist of a timestamp, food product identification number, truck vehicle identification number, amount of transported food, the stage the truck is coming from, and the network address of the previous block. Initially, the data is structured as a dictionary.

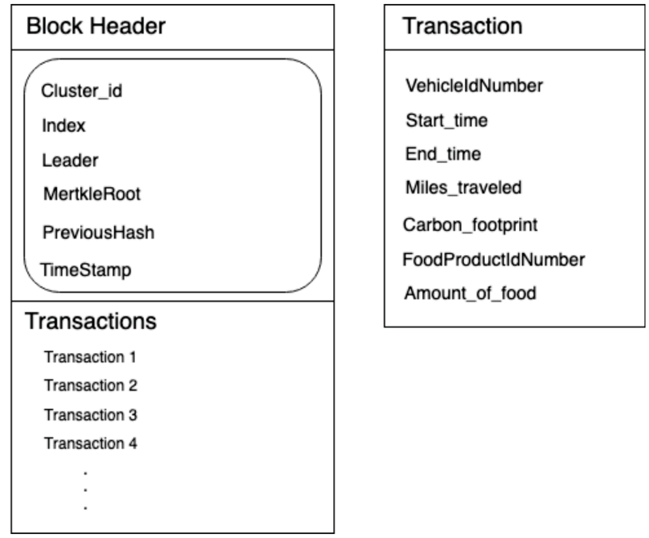

Fig. 2: Block structure.

By applying a one-way hash function (e.g. SHA256) on the data, a blockchain block is created. Figure 2 shows the block and transaction structure. Our implementation utilizes a Merkle tree to store the data in the Blockchain [21]. The Merkle tree summarizes all transactions in a block. Each leaf node is a hash of transactional data, and each non-leaf node is a hash of the child nodes.

D. Blockchain Consensus Algorithm

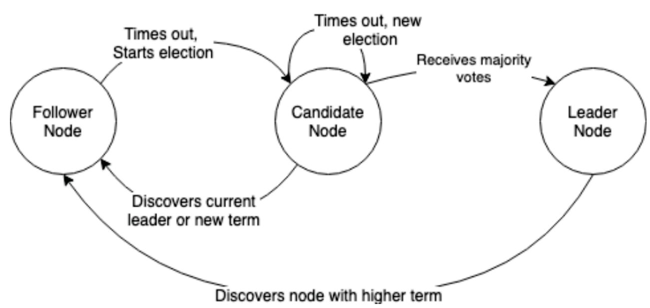

Fig. 3: State diagram of consensus algorithm.

The proposed CFC has three types of nodes: leader, follower, and candidate nodes. A leader is a node that broadcasts the blocks to other nodes. A follower is a node that listens to the broadcast, and a candidate is a node that initiates an election. Figure 3 shows the state diagram of node transitions of the consensus algorithm. Each node has a randomized timer also known as the heartbeat. The node that reaches the end of the timer first broadcasts a state change to a candidate node, and requests a vote of approval to become the leader. The follower nodes approve elected leader unless there is another candidate node requesting an approval. When nodes reach consensus, the elected leader broadcasts a request to the rest of the network, looking for potential contentions from another node that may have a higher term number. We implement a Raft-like consensus algorithm. Different from Raft, the leader node does not have a direct access to the local blockchains on every node. In fact, the leader node only broadcasts a single block of transactions that are replicated across every node's state machine. The blocks inside the state machine of a node are then written to the local blockchain on every node.

```
if not getLeader() then
    return False
end
values = request.POST
if getLeader() is not self.node then
    url = getLeader() + "/transaction/create"
    response = forwardRequest(url, values)
    if response.status then
        return True
    end
    if self.cluster.name is not "Farm" then
        /* get truck's origin and validate the
           block's hash from the random node in
           previous cluster                          */
        prevHash = values["PreviousHash"]
        prevNode = values["PreviousNodes"]
        node = random(PrevNode)
        url = node +"/validate/hash"
        response = forwardRequest(url, prevHash)
        if not response.status then
            return False
        end
        /* currentTransactions is a replicated
           List on the distributed cluster.
           Transactions added to the leader get
           replicated across all nodes.
           sync=True makes the function blocking
           */
        transaction = getVehicleInformation(values)
        self.cluster.currentTransactions.add(transaction,
         sync = True)
    end
end
```

Algorithm 1: Consensus algorithm.

Algorithm 1 shows the detailed implementation of the consensus algorithm. As the truck arrives at a new stage/cluster, it broadcasts its records to the nodes in that cluster. The records passed from the truck between different blocks include: the previous stage or cluster of the truck, the address of the previous block, traveled path, amount of carbon footprint, timestamp, food product identification number, trucks vehicle identification number, and the amount of transported food. The algorithm first checks whether the current cluster is the first cluster and do not have a previous cluster before that. In order to validate the truck and its freight, the algorithm checks where the truck comes from, which is verified by the address of the previous block and the name of the previous cluster. The address of the previous block is used as a key to retrieve the information from the blockchain. The algorithm randomly picks a node from the previous cluster and makes a request to validate whether that block exists or not. If the block exists, the current transaction is replicated across all nodes of the current cluster concurrently. Otherwise, the information is rejected and is not written on the blockchain.

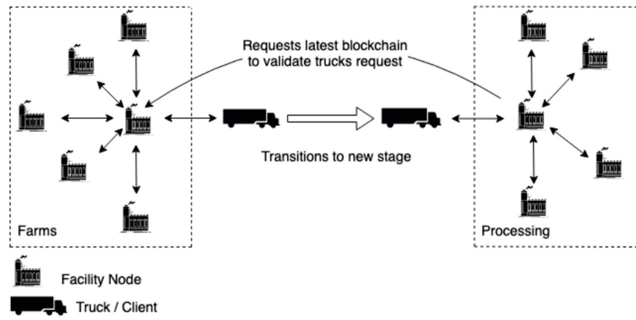

Fig. 4: Communication between clusters.

IV. Experimental Results

We implemented the proposed CFC in Python using Quart and Asynchronous I/O library, *asyncio*. Quart library is used to create a web server that runs the API Endpoints. The experiments were conducted on a Dell tower with Intel Xeon Gold 5122 CPU running at 3.6 GHz and 16 GB of memory, running Ubuntu 18.04.

We simulated the data that come from the HD-OBD and use the web application's Application Programming Interface (API) endpoints to broadcast the data. Each node on the network runs locally on the machine on a separate localhost port.

We evaluated the performance of a CFC with six clusters, each cluster representing a life cycle stage. Each cluster has a different number of nodes and various number of transactions per block. We first experimented with five nodes per cluster and generated one thousand blocks with 5, 10, 20 and 40 transactions in each block to evaluate the block creation time and transaction time.

We also tested the election time for each cluster to elect a leader with 5, 10, and 15 nodes per cluster.

Figure 5 shows the cumulative distribution function (CDF) of block creation time with six clusters using five nodes per cluster and 6006 transactions, including six genesis blocks. The figure shows that 90% of the time, the block creation time is smaller than 0.3 seconds. For reference, note that the Bitcoin network requires around 10 minutes for a single block creation [22].

Figure 6 show the box plot of the distribution of time per transaction within each cluster, for all six stages. Because the first cluster does not need to validate the block of the previous cluster, it takes the smallest amount of time, which is an average of 1.2 seconds per transaction. The time it takes to create a transaction on the other clusters is similar because

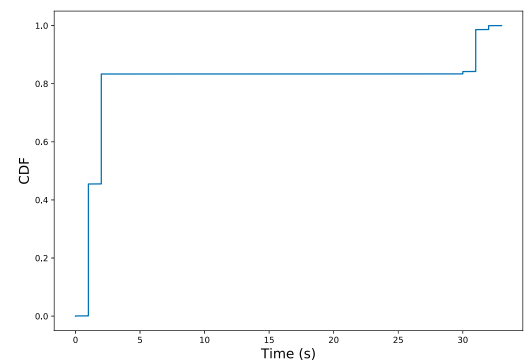

Fig. 5: CDF of block creation time.

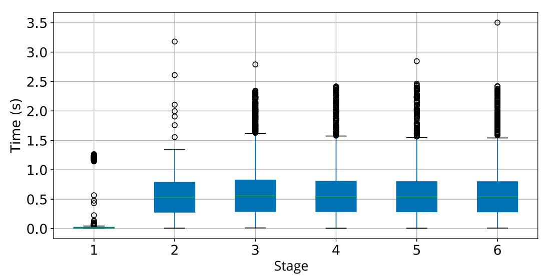

Fig. 6: Time taken per transaction in each cluster.

the clusters only have to check if the block in the previous cluster exists.

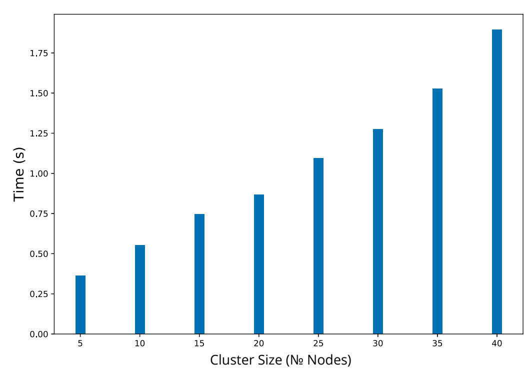

Fig. 7: Leader election time.

Figure 7 shows the average election time as the number of nodes per cluster increases. The election time to elect a leader increases as the number of the nodes increases. For example, with 40 nodes per cluster, it takes about 1.75 seconds to elect a leader.

Table I provides the detailed average transaction time in seconds for each cluster when the number of transactions is 5, 10, 20 and 40. When the number of transactions is smaller than 40, the transaction time is in a few to tens of millisecond range. When the number of transactions increases to 40, the delay increases significantly to hundreds of milliseconds. This may be caused by the resource sharing in a single machine simulation environment.

TABLE I: Average transaction time for each cluster vs. number of transactions

Average transaction time (s)				
Stage	Number of transactions			
	5	10	20	40
1	0.0098	0.0095	0.0092	0.1016
2	0.0128	0.0158	0.0146	0.5593
3	0.0134	0.0141	0.0163	0.7274
4	0.0130	0.0155	0.0164	0.6588
5	0.0154	0.0129	0.0140	0.6616
6	0.0158	0.0152	0.0158	0.6632

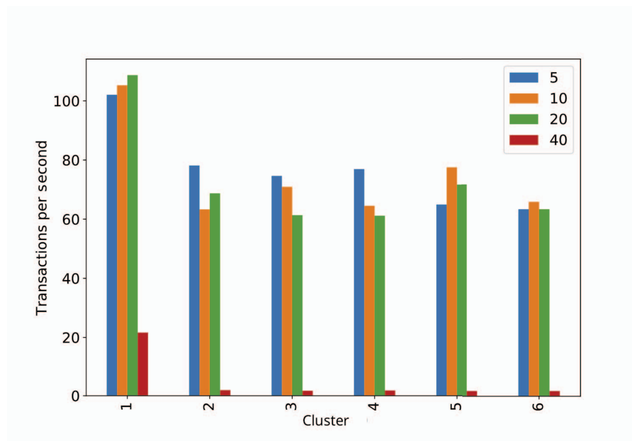

Fig. 8: Cluster throughput vs. cluster size.

Figure 8 shows the throughput for each cluster measured in average number of tps with different cluster sizes (defined as the number of nodes in the cluster). The average throughput for all clusters is above 60 tps when the cluster size is below 40.

Figure 9 shows the storage requirement as the block size (number of transactions) increases. The file size increases when the number of transactions increases. The required storage is less than 4 MB when the number of transactions is 40.

These results show the proposed CFC achieves high performance in terms of block creation time, leader election time, and average transaction time as the blockchain grows as compared to that of a Bitcoin blockchain. CFC can also handle a large number of transactions per second as compared to the traditional blockchain. The cluster-based approach ensures that the block size does not grow exponentially as it occurs in a conventional electronic payment system. Because a transaction is reported only when a truck enters a facility in the next food life cycle stage, the number of concurrent transactions is limited. Evaluating the CFC security is needed

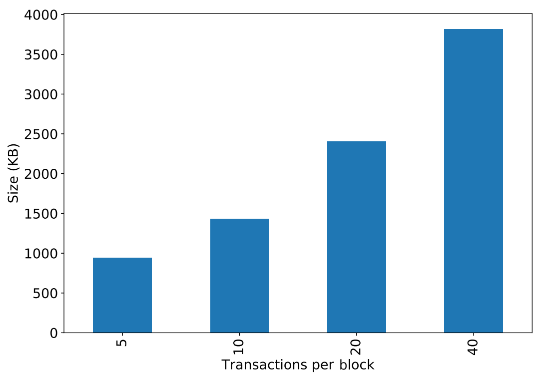

Fig. 9: Required storage vs. number of transactions per block.

to understand the possibility of corrupting a majority of nodes in a cluster with fake transactions. This evaluation is left for future research.

V. Conclusions

We proposed a cluster-based blockchain implementation, which we call the carbon footprint chain, to provide a lightweight distributed-record keeping system for tracking carbon footprint in transportation of food, which is performed when food is transported from one stage to the next in the food life cycle. The proposed carbon footprint chain has six clusters, each of which representing a food life-cycle stage. The facilities within each life cycle stage form the blockchain nodes. Each block contains information about transported goods, carbon footprint, mileage, previous cluster information, etc. Transaction records are added to the block when trucks transport food product from one stage to another. The proposed CFC adopts a Raft-like consensus algorithm by arbitrating decision making on leader election within a cluster, node addition and block updates. We implemented the CFC through web application interface with nodes communicating through a TCP/IP network and evaluated its throughput, latency, and storage. Results show that the proposed CFC has a small block creation time, which is in the order of seconds, and lower average transaction time. The system has a higher throughput with a linearly increase in the amount of storage needed at each block.

Acknowledgment

This project has received funding in the framework of the Joint Programming Initiative Urban Europe, with support from the European Unions Horizon 2020 Research and Innovation Programme under grant agreement No. 730254. This material is based upon work supported by the National Science Foundation Grant Award No. 1830718.

References

[1] "World population prospects, key findings & advanced tables," 2017. [Online]. Available: https://esa.un.org/unpd/wpp/publications/files/wpp2017_keyfindings.pdf

[2] Tilman, D., Balzer, C., Hill, J., and Befort, B. L., "Global food demand and the sustainable intensification of agriculture," *Proceedings of the National Academy of Sciences*, vol. 108, no. 50, pp. 20 260–20 264, 2011. [Online]. Available: https://www.pnas.org/content/108/50/20260

[3] Vsquez, F., Vita, G., and Mller, D. B., "Food security for an aging and heavier population," *Sustainability*, vol. 10, no. 10, 2018. [Online]. Available: http://www.mdpi.com/2071-1050/10/10/3683

[4] Türkay, M., "Environmentally conscious supply chain management," *Transportation*, vol. 1, no. 32, p. 76, 2008.

[5] Weber, C. L. and Matthews, H. S., "Food-miles and the relative climate impacts of food choices in the united states," *Environmental Science & Technology*, vol. 42, no. 10, pp. 3508–3513, 2008. [Online]. Available: https://doi.org/10.1021/es702969f

[6] Wakeland, W., Cholette, S., and Venkat, K., "Food transportation issues and reducing carbon footprint," in *Green technologies in food production and processing*. Springer, 2012, pp. 211–236.

[7] Alkawaleet, N., Hsieh, Y.-F., and Wang, Y., "Inventory routing problem with co 2 emissions consideration," in *Logistics operations, supply chain management and sustainability*. Springer, 2014, pp. 611–619.

[8] Nijdam, D., Rood, T., and Westhoek, H., "The price of protein: Review of land use and carbon footprints from life cycle assessments of animal food products and their substitutes," *Food policy*, vol. 37, no. 6, pp. 760–770, 2012.

[9] Röös, E., *Analysing the carbon footprint of food*, 2013, vol. 2013, no. 56.

[10] Andersson, K., Ohlsson, T., and Olsson, P., "Life cycle assessment (lca) of food products and production systems," *Trends in Food Science & Technology*, vol. 5, no. 5, pp. 134–138, 1994.

[11] Weber, C. L. and Matthews, H. S., "Food-miles and the relative climate impacts of food choices in the united states," *Environmental Science & Technology*, vol. 42, no. 10, pp. 3508–3513, 2008. [Online]. Available: https://doi.org/10.1021/es702969f

[12] Kendall, A., Yuan, J., and Brodt, S. B., "Carbon footprint and air emissions inventories for us honey production: case studies," *The International Journal of Life Cycle Assessment*, vol. 18, no. 2, pp. 392–400, 2013.

[13] "Freight analysis framework," 2019. [Online]. Available: https://ops.fhwa.dot.gov/freight/freight_analysis/faf/index.htm

[14] Roy, P., Nei, D., Orikasa, T., Xu, Q., Okadome, H., Nakamura, N., and Shiina, T., "A review of life cycle assessment (lca) on some food products," *Journal of food engineering*, vol. 90, no. 1, pp. 1–10, 2009.

[15] Scherer, M., "Performance and scalability of blockchain networks and smart contracts," 2017.

[16] Kamath, R., "Food traceability on blockchain: Walmarts pork and mango pilots with ibm," *The JBBA*, vol. 1, no. 1, p. 3712, 2018.

[17] Tian, F., "A supply chain traceability system for food safety based on haccp, blockchain amp;amp; internet of things," in *2017 International Conference on Service Systems and Service Management*, June 2017, pp. 1–6.

[18] Cachin, C., "Architecture of the hyperledger blockchain fabric ," 2016.

[19] Vujičić, D., Jagodić, D., and Ranić, S., "Blockchain technology, bitcoin, and ethereum: A brief overview," in *2018 17th International Symposium INFOTEH-JAHORINA (INFOTEH)*. IEEE, 2018, pp. 1–6.

[20] Chart of the bitcoin blockchain size. [Online]. Available: https://www.blockchain.com/en/charts/blocks-size

[21] Lin, I.-C. and Liao, T.-C., "A survey of blockchain security issues and challenges." *IJ Network Security*, vol. 19, no. 5, pp. 653–659, 2017.

[22] Zohar, A., "Bitcoin: under the hood," *Communications of the ACM*, vol. 58, no. 9, pp. 104–113, 2015.

[23] Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y. *et al.*, "Hyperledger fabric: a distributed operating system for permissioned blockchains," in *Proceedings of the Thirteenth EuroSys Conference*. ACM, 2018, p. 30.

[24] Kwon, J., "Tendermint: Consensus without mining," *Draft v. 0.6, fall*, 2014.

[25] Sprung, M. J. *et al.*, "Freight facts and figures 2017," 2018.

[26] "Public hearing to consider proposed revisions to on-board diagnostic system requirements, including the introduction of real emissions assessment logging (real), for heavy-duty engines, passenger cars, light-duty trucks, and medium-duty vehicles and engines," 2018. [Online]. Available: https://www.arb.ca.gov/regact/2018/hdobd18/isor.pdf