Investigating the link between transaction and computational costs in a blockchain environment

Abdul Jabbar & Samir Dani

Published online: 27 Apr 2020.

Submit your article to this journal

Article views: 706

View related articles

View Crossmark data

Taylor & Francis
Taylor & Francis Group

Check for updates

Investigating the link between transaction and computational costs in a blockchain environment

Abdul Jabbar[a] and Samir Dani[b*]

[a]Director of Learning Development, Department of Logistics, Marketing, Hospitality and Analytics, Huddersfield Business School, University of Huddersfield, Huddersfield, UK; [b]Department of Logistics, Marketing, Hospitality and Analytics Huddersfield Business School, University of Huddersfield, Huddersfield, UK

The research and thinking pertaining to blockchain have thus far focused on cryptocurrency and Bitcoin. However, there is increased interest in using the technology to solve operational challenges in manufacturing and service supply chains. In this study, we introduce a new implication of using blockchain technology and propose two unique contributions. First, we introduce the notion of computational costs (measured in units of gas) as an essential mechanism for completing operational transactions in the blockchain environment. Second, we discuss the use of smart contracts and their influence on operational transactions. To investigate the link between blockchain transaction and computational costs, this study uses an experimental methodology. We develop and implement a fully functional virtual public blockchain to store, validate, and maintain transactions. The methodology provides a process to measure the computational costs, frequency, and intensity of transactions. This research contributes to conceptual research on the blockchain implementation paradigm. Its novelty stems from the identification of computational costs for operational transactions and use of an experimental methodology. This research provides managers an insight into the design of smart contract transactions in a supply chain from a cost perspective.

Keywords: Blockchain; computational costs; operational transaction costs; smart contracts; supply chain

1. Introduction

Industry 4.0 technologies are starting to have a significant strategic impact on applications in manufacturing and service industries, such as scheduling (Dolgui et al. 2019b), and process and globalisation strategies (Stentoft and Rajkumar 2019) to name a few. Blockchain technology has garnered much attention in recent times as a solution for a number of operational challenges in various business sectors. The research and thinking within the blockchain domain are generally focused on cryptocurrency and Bitcoin. There is increased interest in using the blockchain paradigm to solve operational challenges in manufacturing and service industries. In this study, we introduce a new implication of using blockchain technology and propose two unique contributions. First, we introduce the notion of computational costs as an essential mechanism for completing operational transactions in the blockchain environment. Second, we discuss the use of smart contracts and their influence on operational transactions.

These two contributions are based on the work of various influential authors (Roeck, Sternberg, and Hofmann 2019; Stentoft and Rajkumar 2019; Dolgui et al. 2019a; 2019b) who have contributed to the field of Industry 4.0, smart contracts, and blockchains. We identify the aspect of computational costs in a blockchain environment as an exciting emerging area of research that has the potential to alter our views on transaction costs, online business models, supply chain communication speeds, and value-added services to customers. The computational cost concept is rooted in computational power and speed, which are traditional computing architecture issues. However, some researchers in the field view computational costs as a security risk (Cheng et al. 2019). Although this is an emerging concept in the blockchain environment, the view of transaction costs in the interaction between organisations at a computational level is new. While the benefits of blockchain have been discussed at length (Kim and Laskowski 2018), there is little or no discussion on the computational cost as a hidden factor that can increase the cost of public blockchain transactions for buyers and suppliers, especially from an operational transaction perspective.

Roeck, Sternberg, and Hofmann (2019) take an abductive case study approach to study interorganisational ledger technology-based transactions between a buyer and supplier in a physical goods supply chain. We investigate the use of computational costs, which are inherent in transactions completed within a public blockchain, using an experimental

methodology to investigate their importance to Industry 4.0 and operational transaction costs. We define the computational cost as *the overall cost required to complete a transaction in the blockchain*. According to the rules of the Ethereum blockchain, this cost must be paid by the initiator (Wood 2017).

We develop and implement a fully functional virtual public blockchain on a platform known as Ethereum (Wood 2017) to store, validate, and maintain transactions. This methodology provides a process to measure the computational cost, frequency, and intensity of transactions. Ethereum is chosen because of its popularity in the marketplace with both users and businesses. The insights from the contributions are gathered through the analysis of 30 transactions carried out and recorded on the virtual Ethereum blockchain. The extant literature in this domain suggests that the introduction and implementation of blockchain may be a disruptive influence on activities in such areas as production (Dolgui et al. 2019b), logistics (Francisco and Swanson 2018), traceability (Hastig and Sodhi 2019), procurement (White 2017), finance, governance (Shermin 2017), supply chains (Yanling, Eleftherios, and Weidong 2019), and the agri-food industry (Zhao et al. 2019). Blockchain technology also offers a rich environment for academic research. However, most research in this field is still theoretical, with little or no empirical evidence or real-life use cases. Iansiti and Lakhani (2017) suggests that a number of researchers have discussed their concerns about privacy, security, and cost because of the hype around the use and implementation of blockchain.

This research contributes to conceptual research on the blockchain implementation paradigm. Its novelty stems from the identification of computational costs for operational transactions and use of an experimental methodology. The paper is structured in the following manner. Initially, we discuss the emergence of blockchain and the technology from which it emanates. Within this discussion, we discuss the differences between public and private blockchains, the impact of computational costs, and how transactions are added to a blockchain. The importance of smart contracts is discussed with regard to the completion of transactions in an operational environment. The experimental setup and stack (software setup) used to design, develop, and implement our virtual blockchain is discussed at length. This setup forms the testbed to investigate the influence of computational costs. We then discuss our results from the blockchain experiment and future areas of research.

2. Literature review

2.1. Emergence of blockchain

A blockchain is defined as a '*secure distributed ledger which can store and exchange value without the need for traditional intermediaries*'(Cheng et al. 2019). A distributed ledger is defined as '*a technological architecture designed for the clearing and settlement of digital asset trading and distributed computing without having the need for central intermediaries*' (Yeoh 2017). A distributed ledger, the superordinate to a blockchain, was primarily developed as a tool to record Bitcoin transactions, a cryptocurrency developed by Satoshi Nakamoto in 2008 (Ankalkoti and Santhosh 2017). In a blockchain implementation, the method of recording transactions can transform the way organisations conduct business. Dolgui et al. (2019b) view blockchain as a key development towards the fourth industrial revolution, providing technological solutions for the implementation of trust between organisations. This emerging decentralised public distributed ledger is ideal for recording transactions on a medium that is immutable, transparent, and permanent (Mondal et al. 2019; Wang et al. 2019). However, these qualities have created a significant amount of 'hype' around blockchain. Although blockchain may add value in a number of scenarios, the extant literature reports little on the successes and failures of this technology. For instance, Browne (2017) suggests that of the 26,000 blockchain projects active in 2015/16, only 8% are still in use. According to Tapscott and Tapscott (2017), this is a foundational technology that disrupts current thinking, and thus may cause confusion in its application, especially in relation to current market norms and business models.

Some researchers (Schniederjans, Curado, and Khalajhedayati 2020) state that blockchain is still in its infancy and displays all the necessary traits of a disruptive technology, similar to the WWW or introduction of the TCP/IP protocol (Pazaitis, De Filippi, and Kostakis 2017; Sullivan and Burger 2017). Disruptive technologies are usually plagued with issues because of their lack of maturity. This can be caused by a number of factors; however, in the case of blockchain, we attribute this to the complex nature of the platform and lack of a graphical user interface. Thus, it is plagued with performance issues and appeals to only a limited audience (i.e. early adopters). Technology in its early phase often has limited practical applications (Christensen and Bower 1996). The issues with blockchain also extend to technological uncertainty, scalability problems, and costs (Wang et al. 2019). These issues determine uptake and popularity. Schmidt and Wagner (2019) find that blockchain shows symptoms similar to the Internet bubble in 2000, with the looming threat of a bubble burst. In this scenario, organisations with robust business and revenue models will survive. During the last WWW bubble, companies such as Google and Amazon were stronger and more sustainable because of their business models.

The emergence of blockchain has created the potential to disrupt many industries. For example, Min (2019) argues that it has the potential to disrupt organisations and the way they carry out transactions in a similar way to how the MP3 file format disrupted the music industry. This disruption also extends to societal usage, organisational strategy, technological
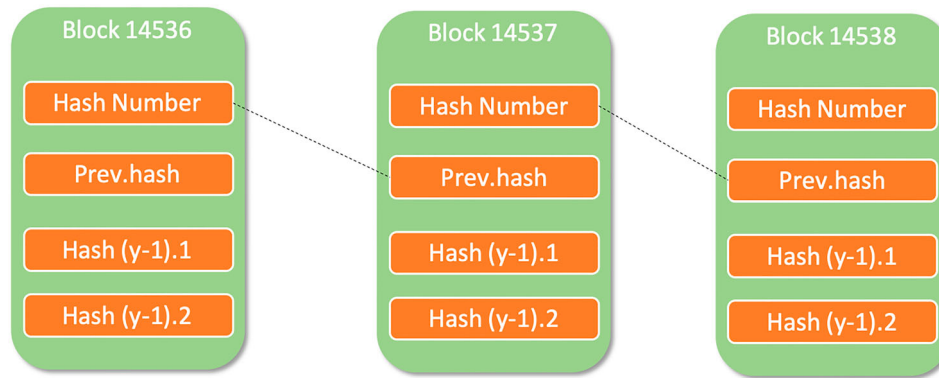
Figure 1. Adding transactions to the blockchain.

innovation, governance, production, efficiency, costs, and trust (Pazaitis, De Filippi, and Kostakis 2017; Sullivan and Burger 2017; Dolgui et al. 2019a; Dolgui et al. 2019b). Within these multitude of applications, there are myriads of industries for which blockchain can be implemented, such as supply chain and manufacturing (Olsen and Borit 2018; Min 2019; Stentoft and Rajkumar 2019; Dolgui et al. 2019a), financial services (Mori 2016; Gai, Qiu, and Sun 2018), food (Zhao et al. 2019), and governance (Shermin 2017).

The deliberate lack of centralisation differentiates a blockchain from a traditional relational database (Jabbar, Akhtar, and Dani 2019), mostly in terms of how individuals and organisations view and store data. This is designed to build trust in systems and ensure that all added transactions are verified by other members of the public blockchain (also known as miners) who control nodes. In principle, this type of permanence approach allows organisations to complete transactions without the need for intermediaries (Yeoh 2017), minimising costs and increasing efficiency. Other studies in this domain consider blockchain technology from an operational perspective. Bai and Sarkis (2020) discuss evaluation methods for selecting the appropriate blockchain technology. Yoon et al. (2020) use a simulation and analytical model to suggest that blockchain technology can be used to manage volatility in international trade. Manupati et al. (2020) focus on monitoring supply chains for carbon emissions and operational costs under a blockchain-inspired modelling approach. Kamble, Gunasekaran, and Arha (2019) conduct an empirical study of the perspectives of blockchain technology adoption using 181 Indian respondents and find a positive view towards blockchain technology adoption.

2.2. Adding transactions to a blockchain

At its core, a blockchain is based on the concept of a traditional relational database (Zhao et al. 2019), with a few significant differences. First, there are considerable differences in the types of data that can be held: structured, semi-structured, and unstructured (Jabbar, Akhtar, and Dani 2019). Second, the decentralisation characteristic changes where the database is located. Finally, owing to its immutability characteristic, a blockchain can act as a permanent record. From an information systems perspective (DeLone and McLean 2003), this may seem to be bad practice, as there is potential for the duplication of data. However, this danger is eradicated using the consensus algorithm and miners.

The process of adding transactions to a blockchain is complex, with multiple actors and significant processing power required to interface in an efficient manner. There are different stages for miners to verify and validate the transactions and computational tasks once these are added into the blockchain (Ankalkoti and Santhosh 2017). For organisations to access or interact with a blockchain, five key steps are involved:

(1) There is an initial call to the blockchain to initiate the transaction.
(2) This transaction is packaged as a block ready for distribution.
(3) The block is then distributed to every party in the network.
(4) Through the process of mining, the network approves this transaction.
(5) The verified block is now added to the blockchain, and a hash that connects the current block to the previous block is generated, thereby creating a chain.

Figure 1 highlights how transactions in a blockchain are linked to each other.

The job of a miner is to select and pool transactions in a block for verification by the other nodes in a network (Ankalkoti and Santhosh 2017). Once verified through a computer algorithm, each block is appended with a hash (Figure 1). The role

of the hash is to maintain the integrity of the block to keep its security, accuracy, and permanency intact (Ankalkoti and Santhosh 2017) before reaching stage five.

2.3. *Public vs. private blockchains*

The adoption of blockchain technology in numerous fields and sectors has continued to grow organically, moving from the financial sector (Mori 2016) into areas such as logistics, property management, and digital identities (Sullivan and Burger 2017). According to Forbes.com (2019), of the top 50 billion-dollar companies exploring blockchain, 50% of current usage of official trials is on the Ethereum platform. According to the same article, investment in blockchain in 2019 was set to hit $2.9 billion. One of the key reasons for the popularity of the Ethereum platform is its ability to create blockchain solutions, which can either be public (Ethereum) or private (Ethereum Alliance).

The development of the Ethereum Alliance in February 2017 by a consortium of major IT and banking organisations placed blockchain technology in the mainstream, creating a level of awareness that increased the popularity of the tool (Anoaica and Levard 2018). The goal of the Ethereum Alliance is to develop a private blockchain environment dedicated to enterprises. The implementation of private blockchains created opportunities for businesses and significantly increased the popularity of public blockchain. The usage of public blockchain then increased, as did the implementation of smart contracts (Anoaica and Levard 2018).

Both private and public blockchains fulfil different roles in the Ethereum ecosystem and have different characteristics. The major difference between them is the basis for permission. A public blockchain by its very nature is 'permissionless': anyone can join the network and read, write, and participate in transactions. This presents decentralisation in action (Pazaitis, De Filippi, and Kostakis 2017). On the contrary, a private blockchain has restrictions on who is and is not allowed to participate and is a 'permissioned' system. The core issue in both these scenarios is how the two systems view users and the role of incentivisation over identity. For private blockchains, and arguably for many industries, it is important to know users (who they are and what they do). This creates certainty and a framework of trust (Anoaica and Levard 2018). Trust is a major cause of concern for most managers who worry about the implications of security on the data integrity of internal and external networks (Wang et al. 2019). To build trust in a public blockchain and minimise fraud and security issues, the focus is on using game theory and incentivisation (cryptocurrency) through mining to encourage good behaviour (Ankalkoti and Santhosh 2017). Both private and public systems have different usage scenarios and consensus algorithms. Research in this area is innovative and many solutions are now being proposed in different contexts. For example, the development of Manuchain is an innovative step forward investigating the use of blockchain in smart manufacturing (Leng et al. 2020). However, to minimise transaction processing time and major bottlenecks, Manuchain must be integrated with other cyber systems (Leng et al. 2020, 184). In addition, Makerchain has been proposed as a blockchain with chemical signatures looking at the self-organising process in social manufacturing (Leng et al. 2019). Both solutions are permission-based and have prototypes based on digital twins. While the source of the blockchain solution proposed by the authors is unclear, the implementation of gas in closed proprietary systems is less of an issue because of the lack of incentivisation scenarios. When private blockchains need access to other actors in an organisation and become 'semi-private', gas as a cost will need to be considered and the consensus algorithm will need to change to incentivise miners.

In both cases, as discussed in this study, computational costs are essential for creating the conditions for blockchain interaction. We argue that a cost is required to add transactions to the block. In this study, we discuss the potential implications of computational costs on transaction costs. Here, a public blockchain is used as part of the experiment because of the popularity of the network, ability to scale, and potential for growth.

2.4. *Smart contracts*

A smart contract, introduced by Nick Szabo in 1994 is defined as a '*set of promises, specified in a digital form, including protocols within which the parties perform on the other promises*' (Antonopoulos and Wood 2018, 127). Since 1994, the notion of smart contracts has evolved significantly from a concept to a usable application, providing opportunities for a multitude of organisations (Dolgui et al. 2019a). The opportunities stem from technologies such as blockchain and cryptocurrency. Smart contracts were introduced to create trust between transactions and eradicate as far as possible intermediaries, which can cause unnecessary expenses and inefficiencies (Christidis and Devetsikiotis 2016). The implementation of blockchain and smart contracts creates a platform for executing transactions based on specific rules and regulations. Contracts are designed to execute when certain conditions and variables have been met (Shermin 2017). A smart contract after deployment resides in a blockchain ready for execution.

Smart contracts, owing to their sensitive nature, are best placed in an environment in which they cannot be changed arbitrarily. A blockchain acts as a permanent state (Castellanos et al. 2017) and does not allow contracts to change without
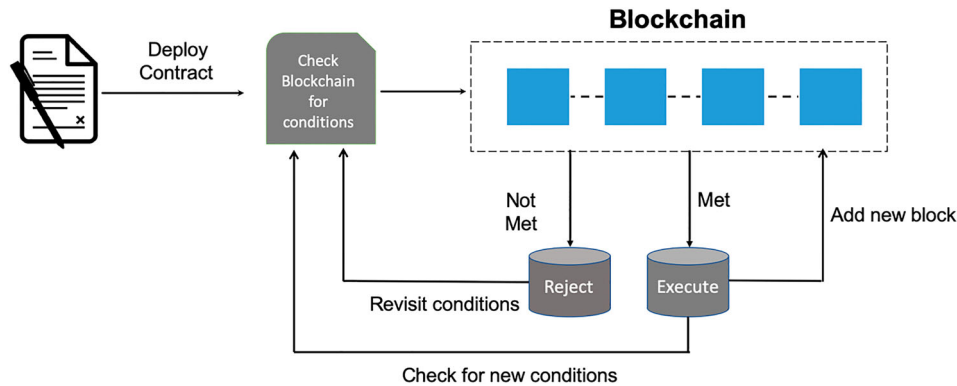
Figure 2. Smart contract relationship with the blockchain.

notifying other members who have a stake in the contract. This permanence creates an environment in which smart contracts can be linked closely to the use of AI in business. This can create an automated system that makes decisions based on rules and regulations in a secure environment. The permanency layer within the blockchain environment provides principles that support the smart contract in the execution of contracts. The aspect of principles has been explored in some fields, especially in law where contracts are designed based on enforceable legal principles (Kosba et al. 2016). This legal enforcement acts as the initial set of rules and regulations that guarantee the execution of contracts and payments (Tapscott and Tapscott 2017). Shermin (2017) suggests that this will be useful in the financial sector where trust-based mechanisms are slow and outdated, based on regulation, contracts, traditions, track records and values, which causes expenses and inefficiency. Figure 2 highlights the execution process during the deployment of a smart contract.

Smart contracts are flexible by nature, and any number of rules and regulations can be programmed as long as they have been mutually agreed (Kim and Laskowski 2018). The ability to programme smart contracts allows for flexibility and creates low overheads, as in many cases these contracts are self-enforcing and automatic, practically eliminating the use of an intermediary and disrupting traditional governance procedures (Shermin 2017). Thus, moving from what is traditionally a manual process to an automated real-time process (Jabbar, Akhtar, and Dani 2019) creates a climate in which the rules and regulations of a contract need to be scrutinised in detail before deployment. Once a contract has been deployed, revising it is impossible because blockchain rules prevent deployed blocks being changed at a later date (Tapscott and Tapscott 2017). Even in scenarios in which a contract change has been agreed, computational costs can be prohibitive. Thus, smart contracts are a key tool in the development and deployment of trust within transactions.

2.5. *Computational costs in a blockchain*

Wood (2017) defines a computational cost as '*the overall cost required to complete a transaction in the blockchain*'. Within the Ethereum environment, a computational cost is viewed as '*a scalar value equal to the number of Wei to be paid per unit of gas for all computational costs incurred as a result of the execution of this transaction*' (Wood 2017). Thus, in the context of the Ethereum blockchain, computational costs, measured in units of gas, are an essential component for the completion and execution of transactions. They play a role in 'lubricating' the network to minimise network traffic and the computational effort needed to complete transactions. The use and implementation of a public blockchain requires significant amounts of energy. According to Stoll, Klaaßen, and Gallersdörfer (2019), the energy required by Bitcoin (which uses a form of public blockchain) adds up to 45.8 TWh, comparable to the energy usage of Jordan and Sri Lanka. These levels of energy consumption place a significant strain on resources and contribute significantly to the cost of mining (Ankalkoti and Santhosh 2017; Stoll, Klaaßen, and Gallersdörfer 2019). The main driver of these high energy costs is its decentralised nature. Owing to the lack of a central bank or governing agency, a public blockchain needs to authenticate itself to verify transactions.

On the Ethereum network, any transaction that requires the use of resources must pay a certain computational cost to overcome this strain on resources. The cost is based on several variables. However, intensity, frequency, and willingness to pay are essential in the calculation. Wood (2017) introduces the following calculation used by the platform to pay miners: *Transaction fee = total gas used * gas price paid (in ether).* In the present study, we use this formula. However, unlike the cost needed to pay miners, we are the first to explore, test, and investigate the costs within operational transactions. Another aspect of this formula not yet tested is the influence of computational intensity and frequency on operational costs. For a

Table 1. Denominations of ether converted into US dollars using https://ethgasstation.info.

Unit	Wei value	Wei	US dollar value
Wei	1 Wei	1	
Kwei (Babbage)	1e3 Wei	1,000	
Mwei (Lovelace)	1e6 Wei	1,000,000	
Gwei (Shannon)	1e9 Wei	1,000,000,000	
Microether (Szabo)	1e12 Wei	1,000,000,000,000	$0.0002
Milliether (Finney)	1e15 Wei	1,000,000,000,000,000	$0.1624
Ether	1e18 Wei	1,000,000,000,000,000,000	$162.39

common smart contract, the cost may be minimal; however, in global transactions with multiple purchase orders, to what extent do computational costs influence operational costs?

The computational cost is viewed as an essential innovation by the blockchain community. This has a significant positive influence on the consensus algorithms 'proof of work' and 'proof of stake' (Mondal et al. 2019), thereby creating a transparent and efficient system for mining operations. The computational cost is a unit of Ethereum and is used by smart contracts to pay miners as part of the consensus algorithm since miners play an essential role in adding and validating transactions to the blockchain. Thus, the pricing of computational costs can vary based on the computational costs used to complete the transaction and price paid (Wood 2017). The final cost is also influenced by the complexity, frequency, and intensity of the smart contract. If an insufficient amount of ether is paid to miners, the contract will fail without completing the transaction and any gas used is refunded. Vitalik Buterin, the creator of the Ethereum platform, based the use of computational costs on the economics of scarcity, which serves as a way to minimise the computational wastage of transactions using unnecessary computing power.

Visualising the computational cost in traditional fiat currency is relatively straightforward when using the Ethereum value on ethgasstation.info. The computational cost is measured in Gwei, and within that the smallest unit of ether is Wei (Wood 2017). 1 ether is equivalent to 1,000,000,000,000,000,000 Wei; thus, this is a very small number to conceptualise in fiat currency. Table 1 shows the various units of ether and how they are valued in US dollars. Adopting the approach of converting ether into US dollars, we calculate the value of transaction costs within the fiat currency. For reference, the value of 1 ether on 24 November 2019 was $150.05 (ethgasstation.info).

We define our work in the literature on transaction cost economics, as discussed by Schmidt and Wagner (2019) and Roeck, Sternberg, and Hofmann (2019). These authors take a conceptual perspective in their research. They focus on how transaction cost theory can be applied to reduce transaction costs and enhance transparency, trust, and disintermediation within blockchain technology. We take a different approach by investigating the influence of the inherent computational costs on operational transaction costs within a buyer–supplier relationship. Thus, we pose two research questions that guide our focus:

RQ1: What is the link between the computational and operational transaction costs within a blockchain?
RQ2: What is the influence of the frequency and intensity of transactions on computational costs?

3. Methodology

3.1. Experimental setup

This research uses the design science research (DSR) strategy introduced by Simon (1996). The use of the DSR strategy for operations management problems has been discussed by Van Aken, Chandrasekaran, and Halman (2016) and Groop et al. (2017). This methodology has also been used by Hukkinen et al. (2019) to consider the Ethereum transaction costs in a blockchain electricity market environment. Based on the DSR strategy, we designed and executed an experiment using the Ethereum virtual machine (EVM) to calculate computational costs. The EVM is a popular blockchain technology for testing smart contracts before deploying them to a live network. It is the standard tool for hosting and executing smart contracts on a simple Turing Complete 256-bit virtual machine (Grech et al. 2018). The use of the EVM fits well with the DSR approach: '*analyse the problem, design a solution, and develop it further in cycles of testing and redesign*' (Van Aken, Chandrasekaran, and Halman 2016). We use Ethereum as the main platform primarily because of its popularity with larger companies testing it as part of their research, such as Google, Amazon, and Citigroup to name a few (Forbes.com 2019).

This setup differentiates itself from that of Roeck, Sternberg, and Hofmann (2019), who use a case study design approach to argue that distributed ledger technology has a significant impact on transaction cost economics, increasing transparency, trust, and disintermediation. We build on this work and embed this within transaction cost theory by investigating the notion of computational costs through the use of smart contracts. To achieve this, we use Solidity (see also Table 3) to develop three
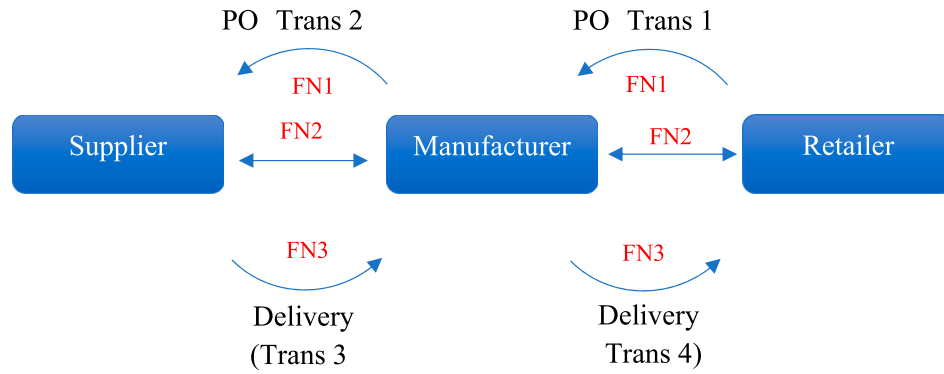
Figure 3. Smart contract functions mapped onto buyer–supplier transactions.

Table 2. List of smart contract functions.

Function ID	Contract name	Purpose of function
FN1	Purchase product ()	The purchase
FN2	Order history ()	Queries the blockchain for all transactions linked to buyers and suppliers
FN3	Sell product ()	Sell when the user is willing to pay the price for the stated number of units

Table 3. Components of the EVM.

Component	Function in stack	Use case scenario
Solidity/Web3.js/Python	Programming languages	Develop, deploy, execute, and update smart contracts
Metamask	Ether wallet	Hold ether for usage in the EVM
Truffle	Development environment	Allows for the execution of smart contracts.
Ganache	Ethereum blockchain	Personal Ethereum blockchain to run tests, execute commands, and inspect states while controlling how the chain operates
Xcode/NPM	Apple development environment	Execute and develop the environment and local webserver

smart contracts to manage the transactions between buyers and suppliers. The experimental scenario presented in Figure 3 shows the basis for depicting the transaction and computational costs. The scenario consists of two dyadic relationships: a retailer–manufacturer relationship and a manufacturer–supplier relationship. This forms a part of the supply chain from the supplier to retailer. The smart contracts in this scenario follow both reverse-pull and forward-push transactions. Table 2 discusses these types of transactions and smart contracts.

Transactions are the purchase orders from the retailer to manufacturer and the manufacturer to supplier. The forward transaction is the movement of goods (delivery) from the supplier to manufacturer and the manufacturer to retailer. This is a simple example of the supply chain; however, it can be useful for depicting transactions within the blockchain environment in real time. Table 2 highlights the three smart contracts and their role in buyer–supplier relationships.

The three functions in Table 2 are responsible for transaction occurrences, recording, intensity, and frequency of contracts. They govern how quickly contracts are executed and when based on conditions. We highlight the role of smart contracts in a simple buyer–supplier relationships and map the smart contracts in purchase orders and transactions.

The EVM has been used as a blockchain resource to avoid adding transactions to the main live environment for several reasons. First, this prevents flooding null transactions in a live space, thereby reducing network overhang. Second, to reduce the cost (ether) of every transaction implemented in a live blockchain, a real-life cost is incurred. In many cases, testing contracts on a live network can become expensive almost instantaneously. Finally, a virtual machine was chosen because of potential security issues under smart contract manipulation (Grech et al. 2018). For clarity, a virtual machine mimics the live blockchain in every way and is thus an accurate measure of costs in a low lag/low cost system. The solutions used to implement the EVM are based on Truffle's suite stack underpinned by Ganache and a local web server. The development and deployment of smart contracts requires five key components as part of the EVM (Table 3).

The components in this table rely on each other for the EVM to function fully. Ganache is a virtual blockchain that provides developers with 100 test ether when linked to Metamask, an online wallet (i.e. a place to store cryptocurrency) used by Truffle to run contracts. Metamask pays the computational cost in the development environment. To view the
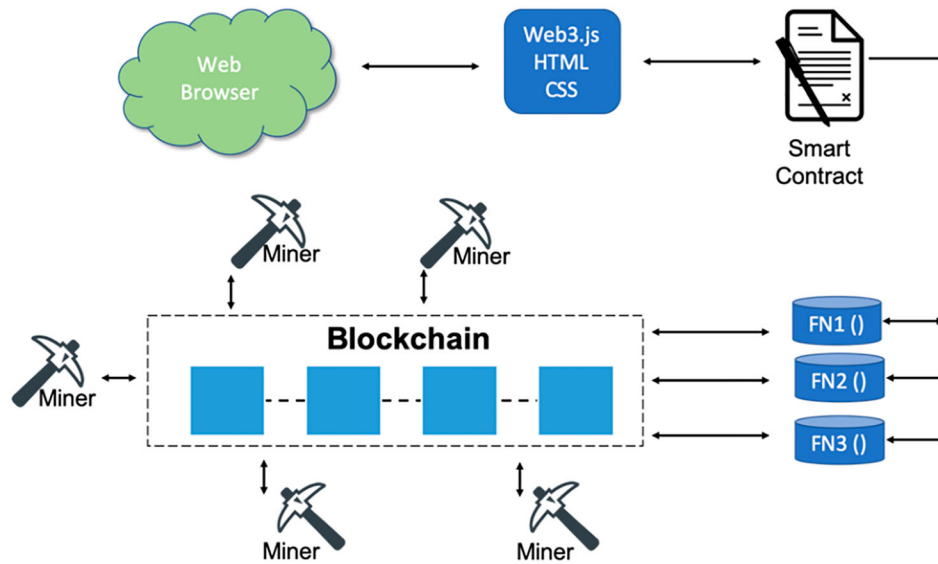
Figure 4. Outline of Truffle's stack.

transaction costs in a live environment, we use Truffle's developer framework tool, alongside the local machine, which we deploy as a local host to test the smart contracts and their influence on costs (Bhargavan et al. 2016). Alongside the Truffle developer environment, the solidity programming language is used as the main programming tool to create smart contracts that can add, maintain, modify, and update the transactions in the blockchain (Bhargavan et al. 2016; Grech et al. 2018). Moreover, we use web3.js to create and deploy the proposed smart contracts. These scripts are essential for controlling the flow of data between buyers and suppliers, which are a key component of a blockchain. In this initiation stage, gas becomes payable (Antonopoulos and Wood 2018).

3.2. Truffle's stack for operational transactions

The developed functions are based on the solidity framework and execution and control of the blockchain. Figure 4 illustrates the stack in full deployment mode.

Using the bytecode stack (Wood 2017; Cheng et al. 2019) shown in Figure 4, we provide a high-level overview of a working blockchain. We highlight the interactions among the actors in the buyer–supplier relationships and propose the notion of computational costs as additional costs that influence transaction costs. For smart contracts to be executed successfully, we initially had to specify the limit for computational cost transactions to initiate and complete on the blockchain. This was a variable rate based on the complexity of the contract (variable contracts provide flexibility to minimise errors and decrease failure rates) (Cheng et al. 2019). However, when deploying a contract for the first time, the limit can be raised on the basis of the block and miner's fees because of the higher levels of intensity required.

4. Results and discussion

4.1. Emergence of computational costs

Our research identified a significant gap in the execution of blockchain transactions, an underexplored area influencing transaction costs. The notion of computational costs, as explained earlier, is based on the formula of Wood (2017), who proposes a dynamic calculation to conceptualise costs in terms of ether and indicate the amount of ether required to be paid to miners for transactions to complete. The formula is essential for virtual machines, especially in relation to testing. Identifying the impact of smart contracts on a virtual machine is a prerequisite for deploying contracts to the live network. Figure 5 illustrates the amount of computational costs used in relation to the type of contract, which shows the value displayed by the EVM.

In this blockchain, the computational cost is variable, depending on the contract type. On the Ganache platform, we identify two types of contracts: *Contract Call* and *Contract Creation*. On the EVM, the *Contract Creation* routine is initiated if a new agreement is reached between two parties (i.e. a buyer and supplier in this scenario). The *Contract Call* routine is an everyday process that runs on the basis of defined user conditions when required. Here, we start to see the permanence

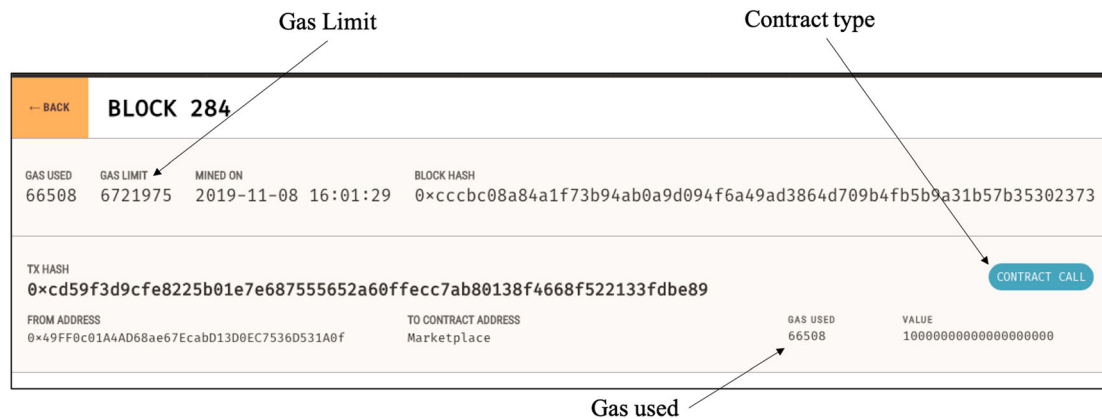Figure 5. Live blockchain running on the Ganache framework.

Figure 6. EVM representation of Wood's (2017) formula.

state of the blockchain; no changes can be made to the *Contract Creation* routine – it cannot be deleted, edited, or removed without the permission of other parties. Indeed, new *Contract Creation* needs to be initiated for any changes to occur. Moreover, as we show in this study, this is an intensive process that incurs high gas costs. Figure 6 shows the computational cost used in the context of the gas limit. Our research suggests that the higher the limit, the higher is the amount of gas used. Hence, intensity plays a key role in this stage.

The two blockchain screenshots in Figure 6 highlight the importance of gas to support the completion of the operation. Four key factors influence the price of the computational cost: the *gas limits, gas used, intensity*, and *frequency*. Through these four factors, we identify the phenomenon of ether depletion. If the necessary gas is not available, the costs can be prohibitive and the failure rate can increase. Thus, to minimise error rates, understanding the four identified factors is crucial for smooth cost transactions. In presenting our data, we argue that the average computational cost varies in relation to the smart contracts deployed, as shown in Table 4.

The average cost may seem small in US dollars, with the average computational cost of a function equating to $0.15–0.30 per transaction. However, this will be influential, as it is dependent on the time required for the transaction and frequency of the transaction. These contracts have low intensity and high efficiency. Therefore, the cost is influenced by how quickly the transaction needs to complete. The data in Table 4 show the link between the average gas used against the gas limit, the price based on a contract, and how often the contract is run. This is an early indicator of the potential implications of the computational cost in transactions between buyers and suppliers. In all cases, in a public blockchain, gas is paid to miners. Miners are then paid in ether on the completion of the task, which is equivalent to the computational effort required

Table 4.　Average costs across the smart contracts.

Smart contract	Avg gas used (Gwei)*	Avg gas limit (Gwei)*	Avg gas limit (USD)*	Avg gas used (USD)*
Contract FN1 ()	176382	937433	$1.39	$0.27
Contract FN2 ()	191116	546047	$0.82	$0.28
Contract FN3 ()	109603	156539	$0.24	$0.17

*Prices correct on date accessed (24 November 2019). Gas is traded on various exchanges, and the value of gas can change daily.

to complete a transaction. Hence, the quicker the transaction, the more processing power is required. This in turn leads to higher transaction fees, as shown in Table 4.

To represent the data at a local level and potential cost implications, we assume that two transactions are run per hour with two shifts over 14 h. We estimate at least 28 transactions as the minimum per contract. Function 1 (FN1) for small business transaction costs is $7.56 on average. The average cost of function 2 (FN2) is $6.72, and that of function 3 (FN3) is $4.76. However, as shown in Figure 3, when FN1 is initiated, this is directly linked to FN3, that is, they run in tandem when specific rules and regulations coded into the smart contract have been met. As a result, the average cost to run 28 transactions is $12.32 per day. Only FN2 is run on demand to check the history. Therefore, the costs of that contract can be controlled, as demonstrated in Figure 3.

The costs calculated for the transactions are based on the experimental setup. An experimental design can generate both random errors and systematic errors. To reduce the systematic errors, we have consulted with blockchain experts when designing the EVM. To reduce random errors, we have conducted 30 transactions and used an average value to showcase the computational cost. The research is designed to show the existence of certain costs that are hitherto unknown within the manufacturing and supply chain domain when using blockchain. Future research will utilise this insight to develop models and test hypotheses in regards to computational costs, intensity, and frequency. The experiment will need to be validated for type1 and type 2 errors, which is currently out of scope for this paper.

4.2. Smart contract challenges

As mentioned by Dolgui et al. (2019a), a blockchain is a platform used to execute smart contracts, a logic-based approach to trust building. The key challenge for organisations in this scenario is estimating the complexity and efficiency of smart contracts to identify potential costs. In this study, even simple contracts that require little or no processing power cost, on average, approximately $0.15 depending on the market conditions. The experiment highlighted some issues in the execution and use of smart contracts. The role of a smart contract in this scenario is to control the flow of information and execute contracts based on set conditions being met. Based on the design of the supply chain and number of transactions (in this case, smart contracts to be executed), the frequency affects the costs. The excessive use of smart contracts can clog networks and lead to issues related to scaling methods. For this reason, computational costs are necessary to protect the network from unnecessary traffic (Mondal et al. 2019).

As indicated in this study and evidenced through the data collection, the verification of smart contracts is a high-value task that has implications for all key stakeholders (Ankalkoti and Santhosh 2017; Stoll, Klaaßen, and Gallersdörfer 2019). However, the unique mix of financial value and recompense for stakeholders makes this an enticing opportunity for unethical behaviour and for the discovery of coding exploitation within smart contracts (Grech et al. 2018). The implementation of computational costs is designed to minimise wasting network resources in executing transactions. Moreover, since this cost is paid up front, the potential of an exception error or an 'out of gas error' increases alongside the likelihood of aborting a transaction. Within this type of scenario, smart contracts are at risk of gas-focused vulnerability with the incorrect handling of gas conditions, causing a denial of service attack (Grech et al. 2018).

Another challenge that smart contract implementations face is the execution of computational costs and limits in relation to the amount of gas used. In the context of this experiment, a gas limit refers to the maximum computational cost we are willing to spend to complete a transaction. Therefore, the challenge in this context is setting a suitable limit at which the transaction can execute without failure. As indicated by the experiment, the more complex the transaction, the more computational work is required. While limits act as a safety mechanism to stop poorly written code depleting funds, the transfer to US dollars can be volatile. In general, a significant amount of research examines cryptocurrency and the volatility of its value against the US dollar (Cong and He 2019). This volatility can influence the computational cost and value of ether; hence, depending on the market conditions, it can significantly alter the transaction cost on an hourly basis.

5. Conclusions and recommendations for future research

In this study, we make two unique contributions that differentiate this research from similar work in the field. First, we propose and introduce the notion of computational costs and idea of gas to measure the cost of completing a transaction in public blockchains. Second, through our EVM experiment, we discuss the challenges of smart contracts in the context of operational costs. We also discuss the existence of computational costs compared with the US dollar as well as depict a fully functioning blockchain and identify the challenges that organisations face in the development of blockchain and smart contract innovations. We position our findings within the context of our research questions and contribute to the literature on blockchains by introducing transaction costs.

5.1. Revisiting the research questions

RQ1: What is the link between the computational and operational transaction costs within a blockchain?

In the first research question, we attempted to investigate the link between computational and transaction costs. Our investigations through the data collected on the EVM found that gas has the potential to increase transaction costs based on platform usage. There is an influence on how transactions are perceived/executed and under which conditions they are deployed, with the implication that there is potential for a significant cost increase. This notion differs from that of other researchers in the field; for instance, Schmidt and Wagner (2019) contend that a blockchain provides efficiency and significant cost savings to organisations. However, we argue that in terms of the architecture at the micro/usage level, new costs arise and these require consideration when designing supply chain operations using blockchain. These costs may be absorbed by savings in other operational areas.

This research is still in its infancy, and the link between computational and transaction costs requires wider study. Future work will need to examine the impact of efficiency and speed on cost. The focus is currently on the emerging aspects of cost, but research needs a wider remit. To meet this research goal, we argue that organisations intending to implement blockchain in their networks need to consider the impact of computational costs on transactions. In this study, we identified the emergence of computational costs as a new area of enquiry in public blockchain. This can impact strategic decisions on how this should be implemented and within what context. This topic needs to be considered in greater detail, as computational costs in their current form have the potential to further disrupt business and revenue models. They could be used to create new streams of revenue not conceived in earlier blockchain iterations.

RQ2: What is the influence of the frequency and intensity of transactions on computational costs?

Wood (2017) defines a formula for calculating gas costs as the total amount of gas used multiplied by the gas price paid. However, the criteria for defining the total computational cost used in this formula are less clear, much like the weighting of each criterion. In this study, we argue that define the total computational cost demands variables such as the influence of frequency, transaction, and intensity. We identified the need to develop the intensity aspect of the work in more detail. Moreover, this research identified significant future areas of research. In addition, once the baseline cost has been set, frequency is based on how often the transaction is executed and how quickly smart contract rules are deployed. Hence, we argue that both frequency and intensity have a significant impact on the cost of operational transactions.

5.2. Managerial implications

The managerial implications of this research provide a focus for manufacturing and supply chain practitioners to understand the use of smart contracts and costs involved per transaction. The development of blockchain technology, particularly smart contracts, is an important step forward within industry, and this is supported by the literature. This has led to an ever-increasing number of academic and professional publications providing insights into this technology. However, our research argues that the majority of these publications focus on the conceptual and technical aspects of the platform rather than operational paradigms. Various researchers (Shermin 2017; Kamble, Gunasekaran, and Arha 2019; Wang et al. 2019) have conceptually suggested the use of blockchain in supply chain environments. We move the research forward by providing managers with an insight into the regular use of this platform for supply chain transactions using an experimental demonstration. We provide a measure of clarity that outlines the ability to conceptualise operational costs to execute smart contracts. Thus, understanding the role of 'gas' in blockchain operations will help managers to design the number of smart contract transactions for a supply chain within the platform. This perspective is a new contribution within the academic literature on production and supply chains.

5.3.   Limitations and recommendations for future research

In this research, we develop a virtual blockchain to mimic a live environment with all the associated transactions and costs. The development of such an approach is innovative but limited by the constraints of the virtual platform. Issues such as high fluctuations in ether value can have a significant impact on transaction costs. Ether costs change rapidly, often on an hour-to-hour basis, creating significant fluctuations in cost. Although our methodology can study these fluctuations within a virtual system, it is still constrained by the ability to replicate the full impact of the fluctuations. In addition, it is difficult to replicate the bandwidth lag, which is the reduction in processing speed due to other machines competing for resources and miners. In the virtual machine, we do not have to contend with the bandwidth lag, which allows us to complete transactions faster than a live blockchain. Although these issues affect the performance and intended intensity of the experiment, we nevertheless provided valuable insights into computational costs.

However, this is potentially a rich area of research with ample opportunity to develop the theory further. To contribute to this discussion, we identify two main areas of discussion. First, there is scope to investigate the impact of speed on transaction costs. For instance, a variable speed that always aims to complete the transaction in the quickest time can be observed in the current experiment. As a result, we argue that further investigation should focus on low-, medium-, and high-speed transactions and the influence of these different speeds on costs. Second, computational costs have the potential to change how online business models are structured. Within an Ethereum blockchain, gas is charged to the person who initiates the transaction. Therefore, it can be assumed that initiators can be charged to access products and services based on a public blockchain.

Disclosure statement

No potential conflict of interest was reported by the author(s).

References

Ankalkoti, Prashant, and S. G. Santhosh. 2017. "A Relative Study on Bitcoin Mining." *Imperial Journal of Interdisciplinary Research (IJIR)* 3 (5): 1757–1761.

Anoaica, Andra, and Hugo Levard. 2018. "*Quantitative Description of Internal Activity on the Ethereum Public Blockchain*." *2018 9th IFIP international conference on New technologies, Mobility and security (NTMS). IEEE*, 1–5.

Antonopoulos, Andreas M., and Gavin Wood. 2018. *Mastering Ethereum: Building Smart Contracts and Dapps*. Sebastopol: O'Reilly Media.

Bai, C., and J. Sarkis. 2020. "A Supply Chain Transparency and Sustainability Technology Appraisal Model for Blockchain Technology." *International Journal of Production Research*. doi:10.1080/00207543.2019.1708989.

Bhargavan, Karthikeyan, Antoine Delignat-Lavaud, Cédric Fournet, Anitha Gollamudi, Georges Gonthier, Nadim Kobeissi, Natalia Kulatova, et al. 2016. "*Formal Verification of Smart Contracts: Short Paper*." *Proceedings of the 2016 ACM Workshop on programming languages and analysis for security*, 91–96.

Browne, R. 2017. "There Were More Than 26,000 New Blockchain Projects Last Year-Only 8% Are Still Active." *CNBC*, November 9.

Castellanos, J., F. Alejandro, Debora Coll-Mayor, and José Antonio Notholt. 2017. "*Cryptocurrency as Guarantees of Origin: Simulating a Green Certificate Market with the Ethereum Blockchain*." *2017 IEEE international Conference on smart energy Grid Engineering (SEGE). IEEE*, 367–372.

Cheng, Zhen, Xinrui Hou, Runhuai Li, Yajin Zhou, Xiapu Luo, Jinku Li, and Kui Ren. 2019. "*Towards a First Step to Understand the Cryptocurrency Stealing Attack on Ethereum*." *22nd international Symposium on research in Attacks, Intrusions and Defenses ({RAID} 2019)*, 47–60.

Christensen, Clayton M., and Joseph L. Bower. 1996. "Customer Power, Strategic Investment, and the Failure of Leading Firms." *Strategic Management Journal* 17 (3): 197–218.

Christidis, Konstantinos, and Michael Devetsikiotis. 2016. "Blockchains and Smart Contracts for the Internet of Things." *IEEE Access* 4: 2292–2303.

Cong, Lin William, and Zhiguo He. 2019. "Blockchain Disruption and Smart Contracts." *The Review of Financial Studies* 32 (5): 1754–1797.

DeLone, William H., and Ephraim R. McLean. 2003. "The DeLone and McLean Model of Information Systems Success: A Ten-Year Update." *Journal of Management Information Systems* 19 (4): 9–30.

Dolgui, Alexandre, Dmitry Ivanov, Semyon Potryasaev, Boris Sokolov, Marina Ivanova, and Frank Werner. 2019a. "Blockchain-Oriented Dynamic Modelling of Smart Contract Design and Execution in the Supply Chain." *International Journal of Production Research* 57: 1–19.

Dolgui, Alexandre, Dmitry Ivanov, Suresh P. Sethi, and Boris Sokolov. 2019b. "Scheduling in Production, Supply Chain and Industry 4.0 Systems by Optimal Control: Fundamentals, State-of-the-Art and Applications." *International Journal of Production Research* 57 (2): 411–432.

Forbes.com. 2019. "Forbes Releases 'Top 50 Billion-Dollar Companies Exploring Blockchain' — Over Half are Working with Ethereum." *Forbes*, November 1. https://media.consensys.net/forbes-releases-top-50-billion-dollar-companies-exploring-blockchain-over-half-are-working-with-d4f9e44f144d.

Francisco, Kristoffer, and David Swanson. 2018. "The Supply Chain Has No Clothes: Technology Adoption of Blockchain for Supply Chain Transparency." *Logistics* 2 (1): 2.

Gai, Keke, Meikang Qiu, and Xiaotong Sun. 2018. "A Survey on FinTech." *Journal of Network and Computer Applications* 103: 262–273.

Grech, Neville, Michael Kong, Anton Jurisevic, Lexi Brent, Bernhard Scholz, and Yannis Smaragdakis. 2018. "Madmax: Surviving Out-of-Gas Conditions in Ethereum Smart Contracts." *Proceedings of the ACM on Programming Languages* 2: 1–27.

Groop, J., M. Ketokivi, M. Gupta, and J. Holmstrom. 2017. "Improving Home Care: Knowledge Creation Through Engagement and Design." *Journal of Operations Management* 53-56: 9–22.

Hastig, Gabriella M., and ManMohan S. Sodhi. 2019. "Blockchain for Supply Chain Traceability: Business Requirements and Critical Success Factors." *Production and Operations Management*. doi:10.1111/poms.13147.

Hukkinen, T., J. Mattila, K. Smolander, T. Seppälä, and T. Goodden. 2019. "Skimping on Gas – Reducing Ethereum Transaction Costs in a Blockchain Electricity Market Application." *Proceedings of the 52nd Hawaii International Conference on System Sciences*, URI: https://hdl.handle.net/10125/60123, ISBN: 978-0-9981331-2-6.

Iansiti, M., and K. R. Lakhani. 2017. "The Truth About Blockchain." *Harvard Business Review* 95 (1): 119–127.

Jabbar, Abdul, Pervaiz Akhtar, and Samir Dani. 2019. "Real-Time Big Data Processing for Instantaneous Marketing Decisions: A Problematization Approach." *Industrial Marketing Management*. doi:10.1016/j.indmarman.2019.09.001.

Kamble, S., A. Gunasekaran, and H. Arha. 2019. "Understanding the Blockchain Technology Adoption in Supply Chains-Indian Context." *International Journal of Production Research* 57 (7): 2009–2033.

Kim, Henry M., and Marek Laskowski. 2018. "Toward an Ontology-Driven Blockchain Design for Supply-Chain Provenance." *Intelligent Systems in Accounting, Finance and Management* 25 (1): 18–27.

Kosba, Ahmed, Andrew Miller, Elaine Shi, Zikai Wen, and Charalampos Papamanthou. 2016. "*Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts*." *2016 IEEE symposium on security and privacy (SP). IEEE*, 839–858.

Leng, Jiewu, Pingyu Jiang, Kailin Xu, Qiang Liu, J. Leon Zhao, Yiyang Bian, and Rui Shi. 2019. "Makerchain: A Blockchain with Chemical Signature for Self-Organizing Process in Social Manufacturing." *Journal of Cleaner Production* 234: 767–778.

Leng, Jiewu, Douxi Yan, Qiang Liu, Kailin Xu, J. Leon Zhao, Rui Shi, Lijun Wei, Ding Zhang, and Xin Chen. 2020. "ManuChain: Combining Permissioned Blockchain with a Holistic Optimization Model as Bi-Level Intelligence for Smart Manufacturing." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 50 (1): 182–192.

Manupati, V. K., T. Schoenherr, M. Ramkumar, S. M. Wagner, S. K. Pabba, and R. Inder Raj Singh. 2020. "A Blockchain-Based Approach for a Multi-Echelon Sustainable Supply Chain." *International Journal of Production Research* 58 (7): 2222–2241. doi:10.1080/00207543.2019.1683248.

Min, Hokey. 2019. "Blockchain Technology for Enhancing Supply Chain Resilience." *Business Horizons* 62 (1): 35–45.

Mondal, Saikat, Kanishka P. Wijewardena, Saranraj Karuppuswami, Nitya Kriti, Deepak Kumar, and Premjeet Chahal. 2019. "Blockchain Inspired RFID-Based Information Architecture for Food Supply Chain." *IEEE Internet of Things Journal* 6 (3): 5803–5813.

Mori, Taketoshi. 2016. "Financial Technology: Blockchain and Securities Settlement." *Journal of Securities Operations & Custody* 8 (3): 208–227.

Olsen, Petter, and Melania Borit. 2018. "The Components of a Food Traceability System." *Trends in Food Science & Technology* 77: 143–149.

Pazaitis, Alex, Primavera De Filippi, and Vasilis Kostakis. 2017. "Blockchain and Value Systems in the Sharing Economy: The Illustrative Case of Backfeed." *Technological Forecasting and Social Change* 125: 105–115.

Roeck, Dominik, Henrik Sternberg, and Erik Hofmann. 2019. "Distributed Ledger Technology in Supply Chains: A Transaction Cost Perspective." *International Journal of Production Research*. doi:10.1080/00207543.2019.1657247.

Schmidt, Christoph G., and Stephan M. Wagner. 2019. "Blockchain and Supply Chain Relations: A Transaction Cost Theory Perspective." *Journal of Purchasing and Supply Management* 25 (4): 100552.

Schniederjans, Dara G., Carla Curado, and Mehrnaz Khalajhedayati. 2020. "Supply Chain Digitisation Trends: An Integration of Knowledge Management." *International Journal of Production Economics* 220: 107439.

Shermin, Voshmgir. 2017. "Disrupting Governance with Blockchains and Smart Contracts." *Strategic Change* 26 (5): 499–509.

Simon, H. A. 1996. *The Science of the Artificial*. 3rd ed. Cambridge, MA: MIT Press.

Stentoft, Jan, and Christopher Rajkumar. 2019. "The Relevance of Industry 4.0 and its Relationship with Moving Manufacturing Out, Back and Staying at Home." *International Journal of Production Research*. doi:10.1080/00207543.2019.1660823.

Stoll, Christian, Lena Klaaßen, and Ulrich Gallersdörfer. 2019. "The Carbon Footprint of Bitcoin." *Joule* 3 (7): 1647–1661.

Sullivan, Clare, and Eric Burger. 2017. "E-Residency and Blockchain." *Computer Law & Security Review* 33 (4): 470–481.

Tapscott, Don, and Alex Tapscott. 2017. "How Blockchain Will Change Organizations." *MIT Sloan Management Review* 58 (2): 10.

Van Aken, Joan, Aravind Chandrasekaran, and Joop Halman. 2016."Conducting and Publishing Design Science Research: Inaugural Essay of the Design Science Department of the Journal of Operations Management." *Journal of Operations Management* 47-48: 1–8.

Wang, Yingli, Meita Singgih, Jingyao Wang, and Mihaela Rit. 2019. "Making Sense of Blockchain Technology: How Will it Transform Supply Chains?" *International Journal of Production Economics* 211: 221–236.

White, Gareth RT. 2017. "Future Applications of Blockchain in Business and Management: A Delphi Study." *Strategic Change* 26 (5): 439–451.

Wood, Gavin. 2017. "Ethereum: A Secure Decentralised Generalised Transaction Ledger (eip-150 Revision)." *Ethereum Project Yellow Paper* 151.

Yanling, Chang, Iakovou Eleftherios, and Shi Weidong. 2019. "Blockchain in Global Supply Chains and Cross Border Trade: a Critical Synthesis of the State-of-the-art, Challenges and Opportunities." *International Journal of Production Research*. doi:10.1080/00207543.2019.1651946.

Yeoh, Peter. 2017. "Regulatory Issues in Blockchain Technology." *Journal of Financial Regulation and Compliance* 25: 196–208.

Yoon, J., S. Talluri, H. Yildiz, and C. Sheu. 2020. "The Value of Blockchain Technology Implementation in International Trades Under Demand Volatility Risk." *International Journal of Production Research* 58 (7): 2163–2183. doi:10.1080/00207543.2019.1693651.

Zhao, Guoqing, Shaofeng Liu, Carmen Lopez, Haiyan Lu, Sebastian Elgueta, Huilan Chen, and Biljana Mileva Boshkoska. 2019. "Blockchain Technology in Agri-Food Value Chain Management: A Synthesis of Applications, Challenges and Future Research Directions." *Computers in Industry* 109: 83–99.