Information Processing and Management 58 (2021) 102525

Contents lists available at ScienceDirect

Information Processing and Management

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

Multi-service model for blockchain networks ☆

Fátima Leal [a,b,*], Adriana E. Chis [a], Horacio González–Vélez [a]

[a] Cloud Competency Centre, School of Computing, National College of Ireland, Dublin, Ireland
[b] REMIT, Universidade Portucalense, Porto, Portugal

ARTICLE INFO

Keywords:
Service networks
Reliability
Quality of service
Network selection
Ethereum
Business models

ABSTRACT

Multi-service networks aim to efficiently supply distinct goods within the same infrastructure by relying on a (typically centralised) authority to manage and coordinate their differential delivery at specific prices. In turn, final customers constantly seek to lower costs whilst maximising quality and reliability. This paper proposes a decentralised business model for multi-service networks using Ethereum blockchain features – gas, transactions, and smart contracts – to execute multiple services at different prices. By employing the Ethereum cryptocurrency token, Ether, to quantify the quality of service and reliability of distinct private Ethereum networks, our model concurrently processes streams of services at different gas prices while differentially delivering reliability and service quality. This multi-service business model has been extensively tested on five concurrent Ethereum networks with various combinations of gas prices, miners, and regular nodes using a Proof of Authority consensus algorithm and throughput as the evaluation metric. It has exhibited linear scalability, providing increased throughput in high-quality Ethereum networks, *i.e.*, composed of more validator nodes. The results also indicate that different mining prices do not impact the network performance, but networks with more miners had limited scalability and an increased level of trustworthiness and reliability.

1. Introduction

As immutable time-stamped data structures, blockchains implement peer-to-peer networks where participant nodes can verify interactions concurrently using decentralised consensus protocols. Data is stored in blocks that are “chained”, *i.e.* each block knows the hash of the previous block, thus creating a ledger. As each node holds a copy of the entire chain, blocks can only be added into the chain when the network nodes reach a consensus. Moreover, in private blockchain networks, nodes should prove their identity to validate transactions. Since blockchain eliminates the need for a central entity, the risk of malicious data manipulation is negligible.

Blockchains intrinsically nurture decentralised digital transactions by eliminating intermediaries such as agents, brokers, or bankers. They provide a feasible approach to the challenge of reliability in decentralised environments by enabling different entities (*e.g.*, individuals, organisations, machines, *etc.*) to communicate and interact with each other autonomously and safely, ultimately enabling the creation of distinct innovative business models and work processes across society and industry (Berdik, Otoum, Schmidt, Porter, & Jararweh, 2021).

☆ This work has been developed under the auspice of the “Smart Pharmaceutical Manufacturing (SPuMoNI)” research project (www.spumoni.eu) funded from 2019 to 2022 by CHIST-ERA, the Horizon 2020 Future and Emerging Technologies programme of the European Union through the ERA-NET Cofund funding scheme (CHIST-ERA BDSI Call 2017) and the Irish Research Council.

* Corresponding author at: Cloud Competency Centre, School of Computing, National College of Ireland, Dublin, Ireland.
E-mail addresses: fatimal@upt.pt (F. Leal), adriana.chis@ncirl.ie (A.E. Chis), horacio@ncirl.ie (H. González–Vélez).

https://doi.org/10.1016/j.ipm.2021.102525
Received 22 June 2020; Received in revised form 20 January 2021; Accepted 20 January 2021
Available online 3 February 2021

The peer-based validation of blockchain transactions is governed by an incentive-based protocol which rewards peers for their validation and computational work. A blockchain infrastructure is widely considered suitable for software solutions which require: (*i*) trustworthiness; (*ii*) immutability; (*iii*) traceability; and, (*iv*) logic-based processing to trigger algorithms and events, a.k.a *smart contracts*.

Smart contracts are composed of autonomous and distributed methods deployed into a blockchain network. These methods can manage transactions and perform smart decisions. Essentially, it is a logic orchestration among entities, assets, and/or goods which a transaction may involve (Zheng et al., 2020). Thus, a blockchain network can be seen as an infrastructure that can enable multi-service processing through smart contracts and transactions.

As a decentralised blockchain platform introduced by Buterin (2013), Ethereum is widely regarded as one of the most interesting platforms for studying blockchain networks due to its structural properties and ample support for smart contracts (Ferretti & D'Angelo, 2020). Our multi-service model is predicated upon concurrent blockchain networks configured with different gas prices. First, a customer proposes the price which he/she is willing to pay for a given service. Then, the system forwards the transactions to the corresponding network in order to be validated and committed into the chain. Thus, each blockchain network holds different smart contracts providing services at different prices. If a customer proposes an unfair gas price, the transaction is discarded and the service is returned to that customer for verification, since low-cost networks do not contain all smart contract methods.

We have employed the Ethereum cryptocurrency token, *i.e.*, Ether, to quantify the service cost considering the level of quality and reliability of each Ethereum network. We have evaluated the model in terms of performance and scalability to optimise the amount of services that can be processed in parallel with Ethereum, and employed Linear Regression (LR) to calculate the optimal performance of a given private Ethereum network with a fixed number of nodes $\{2, 4, 6, 8, 10, 12, 14, 16\}$. In our approach, the higher the service cost, the better the quality control and reliability of the data obtained from the blockchain network.

The evaluation has been carried out on a scalable private Ethereum network (Leal, Chis and González-Vélez, 2020), assembled as part of the Smart Pharmaceutical Manufacturing (SPuMoNI) European project. SPuMoNI aims to systematically assess and trace all data produced by computerised production systems in pharma environments and, consequently, enable pharma manufacturing lines to intrinsically marshal quality, reliability, and traceability (Leal et al., 2021). While the evaluation has used real pharma manufacturing datasets on a Ethereum private network, it is noted that the approach to multi-service blockchain networks is arguably domain- and network-agnostic.

The network has been initially configured with two validator nodes, a *sine qua non* of the Proof of Authority consensus algorithm, and our multi-service model yielded to a maximum throughput of 1000 transactions per second (on average, 600 transactions per second). Knowing this limit, our multi-service model has been furnished with the ability of adaptively increasing the throughput as more Ethereum validator nodes are included in the network. In addition, we have observed that the performance is independent of the mining gas price employed in a network, *i.e.* networks with the same number of nodes provide a relatively similar throughput. In the proposed multi-service model, the larger the number of nodes, the higher the degree of trustworthiness and reliability.

Therefore, as the main contribution, this paper focuses on the employment of a decentralised authority to manage multi-service networks. As the system prioritises services with fair and better prices, our proposed model analyses the offer and estimates the service price according to the transaction complexity and the corresponding computational effort. Better prices have faster processing ergo multiple levels of service according to economic capacity.

> *Consequently, our contention is that blockchain networks can intrinsically determine the user service level via the amount of gas per transaction, hence segmenting services and users based on the expected transaction cost.*

The rest of this paper is organised as follows. Section 2 presents a literature review covering relevant services that use private Ethereum networks. Section 3 describes the proposed method including the configuration of the network, the smart contract, and the evaluation metrics. Section 4 presents the experimental setup, and the empirical evaluation results and their discussion. Section 5 presents conclusions, analysis, and directions for future work.

2. Related work

The first blockchain-based applications, *i.e.*, Blockchain 1.0., enabled decentralised value transfer for cryptocurrencies. The introduction of smart contracts to power fully decentralised (mostly financial) applications led to Blockchain 2.0. The advent of non cryptocurrency-related distributed ledger applications is currently referred to as Blockchain 3.0 (Di Francesco Maesa & Mori, 2020).

Widely considered a disruptive technology (Khanna & Kumar, 2020), Blockchain 3.0 can arguably modify business operations, processes, services, and business models given its intrinsic potential to articulate differently business logic and added value for a given company. It has been proved to help businesses in authenticating traded goods, facilitating disinter-mediation, and improving operational efficiency (Nowinski & Kozma, 2017). Consequently, different research groups have explored the impact of blockchain technology on business models in different domains including healthcare (Hardin & Kotz, 2021), Industry 4.0 (Putz, Dietz, Empl, & Pernul, 2021; da Rosa Righi, Alberti, & Singh, 2020), payments (Holotiuk, Pisani, & Moormann, 2017), smart cities (Esposito, Ficco, & Gupta, 2021; Hirtan, Dobre, & González-Vélez, 2020), smart vehicles (Oham, Michelin, Jurdak, Kanhere, & Jha, 2021), supply chain (Chang, Chen, & Lu, 2019), fake news detection (Chen, Srivastava, Parizi, Aloqaily, & Ridhawi, 2020), and tourism (Leal, Veloso, Malheiro and González-Vélez, 2020). Moreover, blockchain has been explored to ensure data integrity in computer systems

Table 1
Business model approaches supported by blockchain technology.

Framework	Domain	Private networks	Ethereum	Multi-service networks
Ekblaw, Azaria, Halamka, and Lippman (2016)	Healthcare	✓	✓	
da Rosa Righi et al. (2020)	Industry 4.0	✓	✓	
Putz et al. (2021)	Industry 4.0	✓	✓	
Holotiuk et al. (2017)	Payments	✓	✓	
Hirtan et al. (2020)	Smart cities	✓	✓	
Esposito et al. (2021)	Smart cities	✓		
Chang et al. (2019)	Supply chain	✓	✓	
Leal, Veloso et al. (2020)	Tourism	✓	✓	
Current proposal	Domain agnostic	✓	✓	✓

including cloud storage (Li, Wu, Jiang, & Srikanthan, 2020), Internet of Things (Zhao, Chen, Liu, Baker, & Zhang, 2020), and open-source software repositories (D'Mello & González-Vélez, 2019). Specifically, Li et al. (2020) propose an auditing scheme to ensure completeness and correctness of the data underlying blockchain technology for enhanced reliability in decentralised networks.

Multi-service networks have been previously studied in the context of computer communications (Evans & Filsfils, 2007) and more traditionally in marketing (Grönroos, 1984). Their main goal is to balance the provision of high-quality services to users with the ability to maximise the number of users served concurrently by segmenting services and users based on the expected service (*a.k.a.* service level agreement (SLA)) and the perceived service (*a.k.a.* quality of service (QoS)). However, the key difference in the aforementioned views of multi-service is the perspective to administer and monitor services. While computer communication networks rely on semi-autonomic arbitration based on protocols and pre-defined QoS requirements, marketing networks expect a more customer-centred approach derived from service science. Nonetheless, the formal representation and computational modelling of service networks have remained open problems for the services economy (Maglio, Srinivasan, Kreulen, & Spohrer, 2006). With the advent of on-demand IT services, there has been a clear need to link customer requirements and expectation with automated monitoring and adaptation.

Consequently, two main approaches have been introduced in IT communication service networks: (i) game theory (Han, Niyato, Saad, Basar, & Hjorungnes, 2011; Trestian, Ormond, & Muntean, 2012; Tsiropoulou, Katsinis, & Papavassiliou, 2012) to technically address the network selection problem based on adversarial optimisation for user requirements, pricing, device(s) characteristics, and provider constraints; and, (ii) holistic service-science (Carroll, Helfert, & Lynn, 2014; Carroll, Whelan, & Richardson, 2010; Tamburri & Lago, 2011) which rely on the existence of a central manager to reconfigure and optimise the network to satisfy the needs and requirements of customers, as well as to coordinate SLA and QoS matching. On the one hand, centralisation does not necessarily work well in distributed highly-transactional networks as they rely on unique arbitration. On the other hand, fully decentralised techniques struggle to find an ideal combination of static and dynamic parameters thence employing heuristical multi-objective optimisations. Arguably, the introduction of peer-to-peer validation and consensus – intrinsic characteristics of blockchain networks – could eventually achieve a degree of centralised mediation while providing intrinsic resilience and decentralisation.

One of the main concerns about blockchain-based applications is performance. In this context, several research works have analysed multiple blockchain platforms and their corresponding configuration to obtain the highest number of transactions per second (Baniata, Anaqreh, & Kertesz, 2021; De Angelis et al., 2018; Leal, Chis et al., 2020; Xu et al., 2021). Ethereum has proved to be one of the most popular and best performing cryptocurrency-based blockchain platforms, allowing to implement decentralised and transaction-based systems.

Measured in Ether (ETH)—a cryptocurrency with market value—*gas* underpins Ethereum computations (Hu et al., 2021), since each executed instruction in a smart contract costs gas. Beyond the repudiation of transactions, "running out of gas" can lead to gas-focused vulnerabilities and exceptions, especially in public Ethereum networks (Grech et al., 2018). Furthermore, gas plays a key pecuniary role in Ethereum networks, as miners are assigned an Ether (1 ETH) amount according to their computational effort. Gas is considered crucial in Ethereum networks (Albert, Correas, Gordillo, Román-Diez, & Rubio, 2020), since it:

(a) prevents resource waste by allowing miners to bid for work as proposed by an emitter node;
(b) regulates consumption from customers due to its associated pecuniary value; and,
(c) limits the computational power associated with a given transaction.

From (a) above, one can infer that such a gas bidding process between miners and emitters leads to the fluctuation of gas prices. Consequently, the prediction of gas prices is *per se* complex and some predictive approaches have started to appear in the literature (Werner, Pritz, & Perez, 2020). Nonetheless, the bidding process is similar to QoS-driven consumption models for cloud resources (Sharma, Thulasiram, Thulasiraman, Garg, & Buyya, 2012), where customers are interested in the highest QoS at minimum cost, while providers advertise services and associated prices based on a SLA.

2.1. Specific contributions

Blockchain is creating new market opportunities rewriting the conventional business models. Table 1 depicts common models found in the literature supported by blockchain technologies. Since they can use private networks and Ethereum as its infrastructure, it is evident that blockchain has been explored in multiple domains, mainly for payment and provenance purposes. However, scant research has been carried out on the transformative Ethereum properties for business models, *i.e. we have found no related work associated with multi-service networks exploring the decentralised authority of blockchain networks.*

To address the challenge of exploring multi-service networks using a decentralised authority, this paper introduces a multi-service business model for blockchain. The model relies on Ethereum characteristics such as gas and gas price to match the expected and perceived service levels for customers. The model is composed of concurrent private Ethereum networks, where network nodes supervise and validate the transactions/services using the Proof of Authority (PoA) consensus protocol.

The PoA attributes a reputation to each validator node, thence validator nodes are encouraged to mine transactions correctly, otherwise, they will be associated with a negative reputation. Thus, an Ethereum network which involves more miners/validators provides better reliability. Our model treats each transaction as a service where the customer proposes the price he/she is willing to pay to execute and save their service in the blockchain. The service has different levels of complexity, *i.e.*, units of gas, and different prices, *i.e.*, gas prices. Then, each service is forwarded to the corresponding network according to the price proposed by the customer. The final service cost is a combination of the complexity and the network mining price.

Since concurrent private Ethereum networks exhibit differential levels of quality control and reliability for services, customers who require a high level of reliability for their services should then propose higher prices. To empirically validate this model, we have employed real pharma datasets emanated from the SPuMoNI research project. Pharma is arguably a domain with well-established SLAs which requires strict end-to-end traceability and data integrity with ensured QoS. Moreover, we present a performance analysis of concurrent private Ethereum networks that provides an insight into the number of services per second that can be processed in each Ethereum network.

3. Method

Our multi-service model uses *concurrent* private Ethereum networks, where the concurrent processing of Ethereum transactions represents multiple services with different prices. This concurrency provides not only multiple simultaneous options to any given customer, but also a near real-time ability to select a suitable network – a sine qua non in service networks – as the PoA consensus enables a distributed decision based on the available networks and their characteristics. Additionally, such concurrency does not significantly increase the overall framework complexity, as the consensus algorithm is an intrinsic component of the blockchain network.

Extending the notion of the network selection problem in the literature (Trestian et al., 2012), our selection of an Ethereum network is based on:

1. intrinsic network metrics such as consensus algorithms (*e.g.*, PoA or Proof of Work), block and complexity of the service (gas limit in Ethereum), and number of regular and miner nodes;
2. client/device characteristics which determine access support to allow the communication with the network and execution of (Ethereum) tasks;
3. application requirements defined by the number of transactions and smart contracts; and,
4. customer preferences typically related to budget or the price a customer is willing to pay (gas price in Ethereum), and service quality (*e.g.*, empty blocks).

Ultimately, the final service cost is a combination of the service complexity, *i.e.*, consensus algorithms, the units of gas required to execute the service in the network, and the gas price proposed by the customer. With Ethereum, it is possible to quantify the complexity of the service (units of gas) and the price of each network (gas price using Ethers, the Ethereum cryptocurrency). A brief introduction to such blockchain characteristics in the context of Ethereum and their implications for performance is presented in Appendix A of Leal, Chis et al. (2020).

Fig. 1 illustrates such a model composed of: (*i*) concurrent private Ethereum networks; (*ii*) stream of Ethereum transactions; (*iii*) PoA as consensus algorithm; and (*iv*) smart contracts. In this model, the customer proposes the price that he/she is willing to pay for a service. Then, the service, which is in the form of Ethereum transactions, is forwarded to the corresponding network to be validated and executed using smart contract methods. The proposed model executes services at different prices. The services with more profit are executed by powerful and reliable Ethereum networks, *i.e.*, with more miner nodes. This model aims for a higher network to provide faster speed and better trustworthiness. Therefore, a customer who wants a faster and reliable service should propose a higher price. Unfair services are discarded by the networks and returned to the customer for verification.

3.1. Concurrent private ethereum networks

Ethereum is an open-source blockchain platform with a trustful framework for transaction-based systems (Wood, 2014). It uses a cryptocurrency token called *Ether*, which quantifies the interactions with an Ethereum network through transactions. In this context, *wei* is the smallest denomination of Ether. Each Ethereum transaction requires an amount of *gas*, *i.e.*, the computational effort measurement to execute Ethereum transactions. The proposed model relies on Ethereum transactions to submit multi-services with different prices. Table 2 describes the composition of the concurrent private Ethereum networks. Each network holds a different number of miners willing to validate transactions at different prices. Larger networks will provide higher speed and an increased level of trustworthiness. In this approach, the customer proposes a price to pay for a given service. Then, the service is

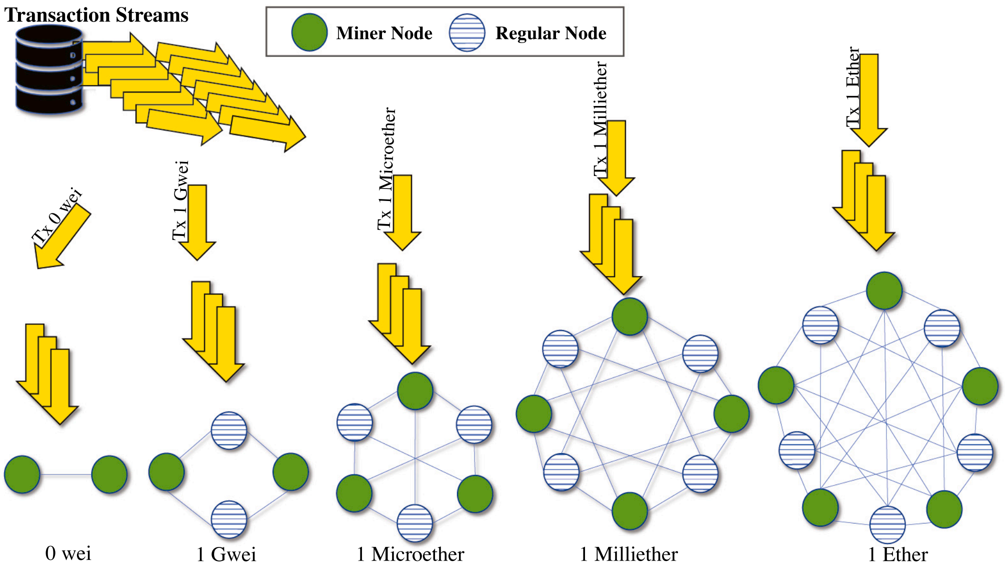

Fig. 1. High-level diagram of the proposed model. It illustrates five private Ethereum service networks configured with a different number of miner and regular nodes – represented as filled and patterned circles respectively – which concurrently execute transactions at five distinct gas prices $\{0\ wei, 1\ Gwei, 1\ Microether, 1\ Milliether, 1\ Ether\}$. The dispatcher handles the distinct streams of transactions/services, represented as arrows, by matching them to a suitable service network according to their proposed price and complexity. *N.B.* The $0 - wei$ network is composed of two miners only due to PoA consensus requirements.

Table 2
wei values and the corresponding number of nodes (regular and miner) for the proposed model.

Network	Unit	Value [wei]	# Regular nodes	# Miner nodes
1	0 wei	0	0	2
2	1 Gwei	10^9	2	2
3	1 Microether	10^{12}	3	3
4	1 Milliether	10^{15}	4	4
5	1 ETH	10^{18}	5	5
…	…	…	…	…
xx	xx ETH	$xx * 10^{18}$	xx	xx

processed and validated by the corresponding network. Regular and miner nodes have different roles. While regular nodes submit the transactions/services to the network, the miners validate and commit the transactions into the chain. The number of miner/validator nodes is always half of the total number of nodes. For example, in a network with six nodes, there are three regular nodes responsible for submitting transactions, and there are three validator nodes.

3.2. Ethereum transactions as multi-services

Composed of a private and a public key, Ethereum accounts incorporate a `gas limit` and a `gas price` to submit instructions known as *transactions*. The `gas limit` indicates the maximum computational effort that a transaction should have to be accepted by the network. Typically, Ethereum transactions have a minimum `gas limit` of 21,000 and a maximum of $6,700,000$ units of gas. The amount of gas per transaction relies heavily on the complexity of the task. Therefore, transactions that require complex interactions with smart contracts use more gas. The `gas price` is the price per unit of gas specified in the transaction by the `gas limit`. Consequently, the final transaction fee is measured as $\texttt{gasprice} * \texttt{gaslimit}$.

Our proposed method converts the transaction structure illustrated in Fig. 2 into services. The gas price corresponds to the service price proposed by the customer and the gas limit to its complexity. The Ethereum account identifies the blockchain network that will process and validate the service. Finally, the contract address identifies the type of service to be executed by the smart contract deployed in the network. The validity of the service is reached by the consensus mechanism of each Ethereum network. The complexity of the service is estimated according to the smart contract needed to execute the service. Thus, unfair service prices will be rejected and returned for verification. The multi-services are forwarded to the corresponding concurrent private Ethereum network in order to be validated and executed simultaneously.

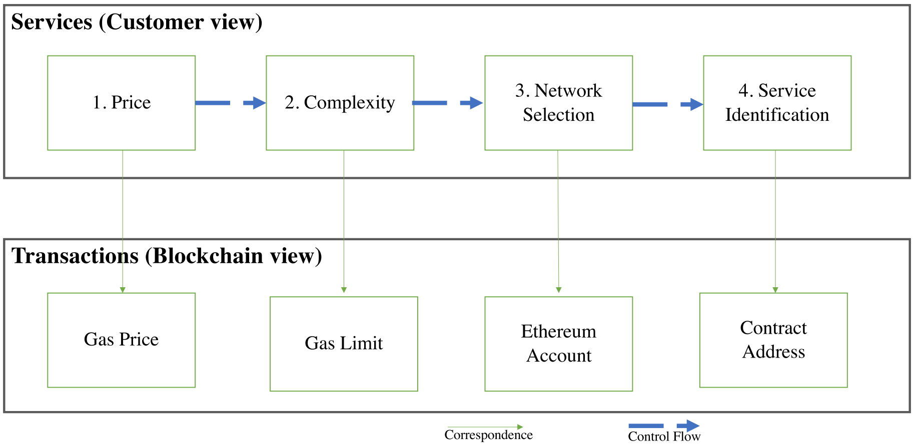

Fig. 2. Service Composition Mapping. It establishes a correspondence between the characteristics of network services and those of blockchain transactions to be executed by the concurrent Ethereum networks. First, a customer proposes the *price* which he/she is willing to pay for a service with a given *complexity*. Then, the system *selects the network* which holds the smart contract to execute the service.

3.3. Consensus mechanism

The multi-services reliability is agreed by all network nodes using consensus mechanisms. Once the consensus is reached, the services are validated, tamper-proof, and included into the chain. The concurrent Ethereum private networks use PoA as a consensus mechanism. PoA uses a reputation-based approach to accept new nodes in a network as validator nodes. A node is able to validate services if at least $\frac{N}{2}+1$ network authorities have previously identified the current node as honest, where N is the number of trusted validators or authorities. Our concurrent private Ethereum networks are composed of a different number of miner/validator nodes according to the network size (see Table 2). Specifically, the PoA works in each network as follows:

– A time slot is allocated to the network node to validate the services. During that time slot the corresponding node is the leader of the network.
– Each node is enabled to validate services every $\frac{N}{2}+1$ blocks. Therefore, the mining frequency of nodes is bounded by $\frac{1}{\frac{N}{2}+1}$.
– A maximum of $N-(\frac{N}{2}+1)$ nodes are allowed to propose blocks in the same time slot.
– The GHOST protocol (Wood, 2014) is applied if multiple nodes are validating the same services at the same time. This protocol gives the validation to the leader.
– The block is sealed by the validator signature.
– A block with multi-services is broadcasted into the network.
– The nodes that constantly propose invalid services reduce their reputation, and consequently can be excluded from the list of reliable validators (De Angelis et al., 2018).

3.4. Smart contracts

Ethereum platforms allow the deployment of smart contracts without any central authority (Singh, Parizi, Zhang, Choo, & Dehghantanha, 2020). Smart contracts enable to build dedicated data structures to accommodate the transactions in the distributed blockchain network. In addition, smart contracts can have multiple functions which enable to perform smart decisions eliminating central authorities. Once deployed in a blockchain network, the smart contracts cannot be deleted or modified.

This paper uses smart contracts to simulate the execution of multiple services. Therefore, the concurrent private Ethereum networks hold different smart contracts allowing to execute services simultaneously at different prices. The smart contracts will allow to allocate, execute, and trace multiple services in a decentralised manner. The proposed model provides services for free, although using limited processing. In turn, profitable services are executed by powerful blockchain networks.

3.5. Performance analysis and evaluation metrics

To evaluate the model, we have analysed the performance using the throughput as an evaluation metric. Throughput is considered a suitable way to measure blockchain network performance, because it quantifies the number of transactions successfully committed in the blocks. We have recorded the throughput and the number of services per block for a given number of services submitted per second to the network, and then employed LR to optimise the performance for multiple numbers of nodes and different gas prices.

Table 3
Dataset composition.

Sources	Transaction description	# Transactions
M1	Asset ID, timestamp, and 6 multiple sensor values	103,403
M2	Asset ID, timestamp, and 6 multiple sensor values	117,788

Throughput. It indicates the number of services per second (sps) successfully committed in the blocks. Eq. (1) depicts the throughput which relies on two constraints: (i) number of services per block; and (ii) block period.[1]

$$sps = \frac{Number\ of\ Services}{Block\ Period} \tag{1}$$

The performance analysis of concurrent private Ethereum networks aims to identify the maximum number of services that a particular blockchain network can receive. In this context, we employ LR to calculate the performance of a network, *i.e.*, the throughput, for a given number of nodes.

Linear regression. LR attempts to model the relationship between an explanatory variable and a dependent variable. In our scenario, we employ LR to model the relationship between the number of nodes N_{nodes} and the throughput sps as depicted in Eq. (2), where β_0 is the intercept and β_1 is the regression coefficient.

$$sps = \beta_0 + \beta_1 N_{nodes} \tag{2}$$

The LR enables to estimate the maximum services per second for private Ethereum networks with a different number of nodes.

4. Experiments and results

To evaluate our model, we performed an empirical evaluation, where we employed: (i) a private Ethereum network with multiple nodes; (ii) multiple transactions which simulate different services; and (iii) smart contracts. Furthermore, our empirical testbed has employed two 3-year time series collected from two distinct pharmaceutical manufacturing lines. The anonymised data encompasses sensor values involved in critical medicine manufacturing. Then, we created multi-services to be executed in the concurrent private Ethereum networks to obtain quality control and the reliability of the medicine manufacturing processes.

The evaluation of our model focuses on concurrent private Ethereum networks with different numbers of nodes and different gas prices. We conducted empirical evaluations for the following two scenarios: (i) maximum performance of a private Ethereum network free of charge, *i.e.*, at 0 gas price; and (ii) concurrent private Ethereum networks with multiple nodes and at different prices. These experiments use a block period of 2 s and a block gas limit of `0x10000000`.

The experiments were conducted on OpenStack cloud instances with 16 GB RAM, 8 CPU and 160 GB of hard-disk space. The implementation has been tested and deployed in the OpenStack platform with Java Development Kit version 13.0.2, Solidity version 0.6.7, Web3J version 4.5.18, and Geth version 1.9.13.

4.1. Pharma dataset and smart contracts

To evaluate the model, we employed a real-world pharmaceutical dataset which is composed of values captured by sensors from two machines (M1 and M2) from pharmaceutical manufacturing product lines. The machines incorporate temperature, pressure, and velocity sensors. Table 3 provides a description of the dataset composition.

Moreover, we create different smart contracts that accommodate and execute the services which are in the form of Ethereum transactions. Specifically, each blockchain transaction/service is composed of: (i) an asset identifier (ID); (ii) timestamp of the sensors' data records; (iii) two values for the temperature sensors; (iv) one value for the pressure sensor; and (v) three values for the multiple velocity sensors.

The pharmaceutical industry requires auditable computerised systems to regulate the product quality. Since traceability and reliability are two main features of the blockchain technology, the pharmaceutical domain is an ideal domain to evaluate the emerging Blockchain 3.0 applications/models. In this context, we use the pharmaceutical data collected from a real production manufacturing line at a European pharma manufacturer to evaluate our model. Using this dataset, we developed smart contracts that enable to store the data and audit the sensor values. In this model, each customer proposes the price that he/she is willing to pay. Then, the services are forwarded and executed to the corresponding Ethereum network. The model offers decentralised reliability according to the price proposed by the customer. The final price is a combination of the complexity of the transaction/service and the mining price of the network. With this data, we have carried out a performance evaluation and analysis of concurrent private Ethereum networks.

[1] **N.B.** This paper has adopted a block period of 2 s for all experiments.

Table 4
Free of charge private ethereum network performance, *i.e.*, at 0 gas price.

Submissions (sps)	# Transactions/block	(sps)	# Empty blocks
78	151	75	0
164	325	161	0
323	654	327	0
476	902	450	0
566	1133	565	0
588	2248	576	✓
600	2343	590	✓

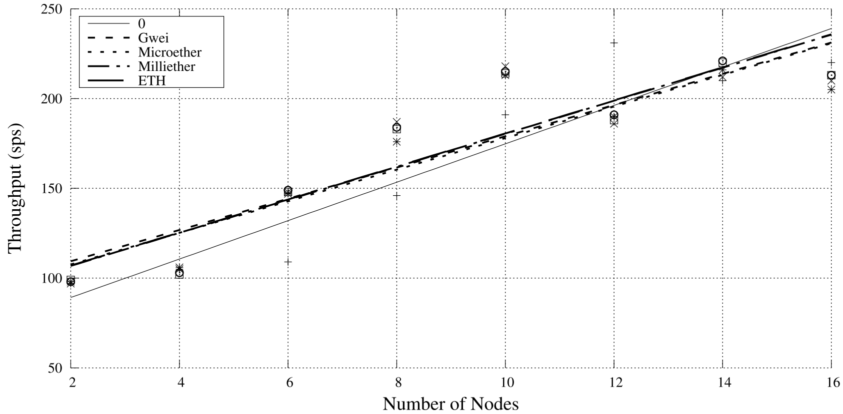

Fig. 3. Analysis of the network behaviour using multiple nodes and different gas prices. It is noted that the throughput is expressed in sps (services per second). The mining gas price does not influence the performance of the network.

4.2. Free of charge private ethereum network performance

A free service, normally, offers the minimum or a limited number of functionalities or requests. Our model encompasses a dedicated private Ethereum network that offers free services. However, this network is composed of only two nodes, offering a low level of reliability. Table 4 contains the performance results of a private Ethereum network with two nodes that are free of charge, *i.e.*, with 0 wei as the mining gas price. The results show that a private Ethereum network with these characteristics accepts on average a maximum of 565 sps. At a value higher than that, the network starts to mine empty blocks and the nodes stop the services broadcasting in the network. In the proposed model, we configured the smallest network to receive until 100 sps.

4.3. Concurrent private Ethereum networks performance

The concurrent private Ethereum network will receive multi-services at the same time and with different prices. As a decentralised technology, the services are managed by a distributed authority using consensus algorithms. In our case, we employ PoA to ensure that all nodes are known and trustworthy. In addition, the network reliability improves as the number of miners increases. Therefore, lower priced networks offer services with less quality and reliability. However, as the network grows it becomes harder to achieve the consensus, and, consequently, the performance decreases. Table 5 contains the performance results of the concurrent private Ethereum networks. Our approach uses half of the network nodes as service submitters and the remaining as miners, *i.e.*, service validators. This approach works up to 220 sps. At a value higher than 220, the network rejects some transactions when the consensus is not achieved. In addition, we can notice that the mining gas price does not influence the performance of the network.

Fig. 3 depicts the network behaviour with multiple nodes and at different gas prices. The degree of trustworthiness and reliability increases according to the number of miners in each network. Therefore, if a customer wants to execute a service with a high level of trustworthiness and reliability they should pay more. We can conclude that the mining gas price does not have a big impact on a private Ethereum network performance presenting similar behaviours.

Our model intends to scale-up the number of services as the network grows. With this purpose, we employ LR which allows to estimate the throughput of each network according to the number of nodes. Table 6 includes the LR results for multiple gas prices.

The proposed model enables a customer to propose the price paid for the service. Then, the service is executed in the corresponding Ethereum network. Therefore, the final cost combines the service complexity, *i.e.*, the units of gas required to execute the service in the network and the mining gas price of each network. Table 7 contains examples of service costs concerning the different concurrent private Ethereum networks. The selection of network is done analysing the service complexity and the price proposed by the customer.

Table 5
Behaviour of the private Ethereum network using different numbers of nodes. The experiments were conducted with different mining gas prices.

Nodes	Submissions (sps)	# Transactions/block	(sps)	Gas price
2	97	147	97	0 wei
	97	144	97	1 Gwei
	98	196	98	1 microether
	99	198	99	1 milliether
	98	196	98	1 ETH
4	107	212	106	0 wei
	106	212	106	1 Gwei
	105	210	105	1 microether
	104	203	102	1 milliether
	104	206	103	1 ETH
6	110	218	109	0 wei
	148	301	148	1 Gwei
	149	294	147	1 microether
	150	298	148	1 milliether
	150	300	149	1 ETH
8	146	293	146	0 wei
	189	374	187	1 Gwei
	187	352	176	1 microether
	190	362	183	1 milliether
	187	368	184	1 ETH
10	193	383	191	0 wei
	217	430	218	1 Gwei
	219	428	213	1 microether
	219	429	214	1 milliether
	218	431	215	1 ETH
12	236	461	231	0 wei
	194	372	186	1 Gwei
	197	380	190	1 microether
	195	375	188	1 milliether
	198	381	191	1 ETH
14	213	429	210	0 wei
	211	440	212	1 Gwei
	220	435	216	1 microether
	220	440	220	1 milliether
	222	442	221	1 ETH
16	220	440	220	0 wei
	215	435	210	1 Gwei
	210	422	205	1 microether
	213	426	213	1 milliether
	220	430	213	1 ETH

Table 6
LR results. The linear regression allows to estimate the amount of *sps* a network with a given number of nodes can execute for the multiple gas prices.

Gas price	LR	R^2
0 wei	$sps = 10.7\ N_{nodes} + 67.8$	0.88
1 Gwei	$sps = 8.7\ N_{nodes} + 92$	0.79
1 Microether	$sps = 8.8\ N_{nodes} + 90$	0.83
1 Milliether	$sps = 9.2\ N_{nodes} + 88.4$	0.83
1 ETH	$sps = 9.2\ N_{nodes} + 88.7$	0.83

Table 7
Example of computation of the total service cost by the concurrent private Ethereum networks.

Network mining cost	Service complexity	Total service cost
0 wei	30,000 units of gas	0 ETH
1 Gwei	40,000 units of gas	0.000004 ETH
1 Microether	50,000 units of gas	0.05 ETH
1 Milliether	60,000 units of gas	60 ETH
1 ETH	70,000 units of gas	70,000 ETH

5. Conclusions

Blockchain technologies have been increasingly employed to address the challenge of provenance in decentralised networks, since data-intensive distributed environments are prone to manipulation when managed by a central entity. Furthermore, blockchain provides data quality control in the form of data authenticity and traceability. Moreover, it introduces the concept of decentralised authority through consensus mechanisms.

In this context, this paper has explored the decentralisation of multi-service networks using blockchain. Typically, multi-service networks provide different services in the same infrastructure which is managed and coordinated by a central entity. Therefore, this paper attempts to decentralise multi-service network models by using blockchain-based SLAs. In this context, we contribute with a decentralised business model which executes multiple services using concurrent private Ethereum networks configured with different mining prices. Each network offers different levels of reliability and quality control depending on the number of nodes that form the network. The higher the number of nodes, the better the quality control and reliability. As a consensus mechanism, we have used PoA which ensures that the service validators are reliable.

The services are processed concurrently by the different Ethereum networks at different prices. The services are in the form of Ethereum transactions where: (*i*) the gas limit represents the complexity of the service; (*ii*) the gas price represents the price that a customer is willing to pay for the service; and (*iii*) smart contracts hold the methods which will execute the service. The final service cost is a combination of the service complexity, *i.e.*, the units of gas required to execute the service in the network, and the gas price proposed by the customer.

Although our proposed model is domain agnostic, it has been evaluated using real pharmaceutical-related datasets collected from manufacturing lines of a pharmaceutical company. Pharmaceutical manufacturing lines require strict end-to-end traceability and data integrity in the medicine production. Therefore, in this case, the concurrent Ethereum networks provide different levels of quality control, traceability, and reliability to the pharma data according to the price proposed by the customer. To evaluate this approach, we have carried out a performance analysis of the concurrent private Ethereum networks. The results indicate that different mining prices do not impact the network performance.

In addition, we have performed a scalability analysis by systematically increasing the number of submitter nodes in each network. In this context, we have employed a LR analysis which enables to estimate the throughput of each network according to the number of nodes utilised.

As future work, we intend to: (*i*) articulate a hybrid multi-level service model by exploring different mining prices in distinct Ethereum networks; (*ii*) enrich the data control mechanisms via smart contract methods; and (*iii*) improve the performance by exploring other Ethereum clients.

In addition, in near future work, we aim to explore the applicability of our multi-service model in real pharmaceutical-related environments. We believe it will enable to automatically provide distinct SLAs with measurable QoS while maintaining reliability and transparency to an industry where compliance and risk assessment are fundamental.

CRediT authorship contribution statement

Fátima Leal: Investigation, Software, Methodology, Formal analysis, Validation, Writing - original draft. **Adriana E. Chis:** Investigation, Funding acquisition, Supervision, Writing - review & editing. **Horacio González–Vélez:** Conceptualization, Funding acquisition, Resources, Writing - review & editing.

Declaration of competing interest

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

Acknowledgement

The kind assistance of the technical team at IDA-Fareva – a core member of the SPuMoNI consortium – is hereby acknowledged.

References

Albert, E., Correas, J., Gordillo, P., Román-Diez, G., & Rubio, A. (2020). GASOL: Gas analysis and optimization for Ethereum smart contracts. In *LNCS*: *vol. 12079*, *TACAS 2020* (pp. 501–506). Dublin: Springer, http://dx.doi.org/10.1007/978-3-030-45237-7_7.

Baniata, H., Anaqreh, A., & Kertesz, A. (2021). PF-BTS: A privacy-aware fog-enhanced blockchain-assisted task scheduling. *Information Processing & Management*, *58*(1), Article 102393. http://dx.doi.org/10.1016/j.ipm.2020.102393.

Berdik, D., Otoum, S., Schmidt, N., Porter, D., & Jararweh, Y. (2021). A survey on blockchain for information systems management and security. *Information Processing & Management*, *58*(1), Article 102397. http://dx.doi.org/10.1016/j.ipm.2020.102397.

Buterin, V. (2013). *A Next-Generation Smart Contract and Decentralized Application Platform*: *Revision 177–9/Dec/2020 Update Ethereum Wiki*, Online at https://ethereum.org/en/whitepaper/ (Last Accessed: 5/Jan/2021).

Carroll, N., Helfert, M., & Lynn, T. (2014). Towards the development of a cloud service capability assessment framework. In Z. Mahmood (Ed.), *Computer Communications and Networks Series*, *Continued Rise of the Cloud* (pp. 289–336). London: Springer, http://dx.doi.org/10.1007/978-1-4471-6452-4_12, chapter 12.

Carroll, N., Whelan, E., & Richardson, I. (2010). Applying social network analysis to discover service innovation within agile service networks. *Service Science*, *2*(4), 225–244. http://dx.doi.org/10.1287/serv.2.4.225.

Chang, S. E., Chen, Y.-C., & Lu, M.-F. (2019). Supply chain re-engineering using blockchain technology: A case of smart contract based tracking process. *Technological Forecasting and Social Change*, *144*, 1–11. http://dx.doi.org/10.1016/j.techfore.2019.03.015.

Chen, Q., Srivastava, G., Parizi, R. M., Aloqaily, M., & Ridhawi, I. A. (2020). An incentive-aware blockchain-based solution for internet of fake media things. *Information Processing & Management*, *57*(6), Article 102370. http://dx.doi.org/10.1016/j.ipm.2020.102370.

De Angelis, S., Aniello, L., Baldoni, R., Lombardi, F., Margheri, A., & Sassone, V. (2018). PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain. In *CEUR Wksp.: vol. 2058*, *ITASEC 2018* (pp. 1–11). Milan: CEUR-WS.

Di Francesco Maesa, D., & Mori, P. (2020). Blockchain 3.0 applications survey. *Journal of Parallel and Distributed Computing*, *138*, 99–114. http://dx.doi.org/10.1016/j.jpdc.2019.12.019.

D'Mello, G., & González-Vélez, H. (2019). Distributed software dependency management using blockchain. In *PDP 2019* (pp. 132–139). Pavia: IEEE, http://dx.doi.org/10.1109/EMPDP.2019.8671614.

Ekblaw, A., Azaria, A., Halamka, J. D., & Lippman, A. (2016). Medrec: Using blockchain for medical data access and permission management. In *OBD 2016* (pp. 25–30). Vienna: IEEE, http://dx.doi.org/10.1109/OBD.2016.11.

Esposito, C., Ficco, M., & Gupta, B. B. (2021). Blockchain-based authentication and authorization for smart city applications. *Information Processing & Management*, *58*(2), Article 102468. http://dx.doi.org/10.1016/j.ipm.2020.102468.

Evans, J. W., & Filsfils, C. (2007). Deploying IP and MPLS qos for multiservice networks: Theory and practice. In *The Morgan Kaufmann Series in Networking*, San Francisco: Morgan Kaufmann, ISBN: 0123705495.

Ferretti, S., & D'Angelo, G. (2020). On the Ethereum blockchain structure: A complex networks theory perspective. *Concurrency Computations: Practice and Experience*, *32*(12), Article e5493. http://dx.doi.org/10.1002/cpe.5493.

Grech, N., Kong, M., Jurisevic, A., Brent, L., Scholz, B., & Smaragdakis, Y. (2018). MadMax: Surviving out-of-gas conditions in Ethereum smart contracts. *Proceedings of the ACM on Programming Languages*, *2*, 1–27. http://dx.doi.org/10.1145/3276486.

Grönroos, C. (1984). A service quality model and its marketing implications. *European Journal of Marketing*, *18*(4), 36–44. http://dx.doi.org/10.1108/EUM0000000004784.

Han, Z., Niyato, D., Saad, W., Basar, T., & Hjorungnes, A. (2011). *Game Theory in Wireless and Communication Networks: Theory, Models, and Applications*. Cambridge, UK: Cambridge University Press, ISBN: 978-0521196963.

Hardin, T., & Kotz, D. (2021). Amanuensis: Information provenance for health-data systems. *Information Processing & Management*, *58*(2), Article 102460. http://dx.doi.org/10.1016/j.ipm.2020.102460.

Hirtan, L., Dobre, C., & González-Vélez, H. (2020). Blockchain-based reputation for intelligent transportation systems. *Sensors*, *20*(3), 791:1–24. http://dx.doi.org/10.3390/s20030791.

Holotiuk, F., Pisani, F., & Moormann, J. (2017). The impact of blockchain technology on business models in the payments industry. In *WI 2017* (pp. 912–926). St. Gallen: AIS.

Hu, T., Liu, X., Chen, T., Zhang, X., Huang, X., Niu, W., et al. (2021). Transaction-based classification and detection approach for Ethereum smart contract. *Information Processing & Management*, *58*(2), Article 102462. http://dx.doi.org/10.1016/j.ipm.2020.102462.

Khanna, P., & Kumar, S. (2020). Engineering 4.0: Future with disruptive technologies. In R. da Rosa Righi, A. Alberti, & M. Singh (Eds.), *Blockchain Technology for Industry 4.0* (pp. 131–147). Singapore: Springer, http://dx.doi.org/10.1007/978-981-15-1137-0_7.

Leal, F., Chis, A. E., Caton, S., González-Vélez, H., García-Gómez, J. M., et al. (2021). Smart pharmaceutical manufacturing: Ensuring end-to-end traceability and data integrity in medicine production. *Big Data Research*, *24*, Article 100172. http://dx.doi.org/10.1016/j.bdr.2020.100172.

Leal, F., Chis, A. E., & González-Vélez, H. (2020). Performance evaluation of private Ethereum networks. *SN Computer Science*, *1*(5), 285:1–17. http://dx.doi.org/10.1007/s42979-020-00289-7.

Leal, F., Veloso, B., Malheiro, B., & González-Vélez, H. (2020). Trust and reputation smart contracts for explainable recommendations. In *Advances in Intelligent Systems and Computing*: *vol. 1159*, *WorldCist'20* (pp. 124–133). Budva: Springer, http://dx.doi.org/10.1007/978-3-030-45688-7_13.

Li, J., Wu, J., Jiang, G., & Srikanthan, T. (2020). Blockchain-based public auditing for big data in cloud storage. *Information Processing & Management*, *57*(6), Article 102382. http://dx.doi.org/10.1016/j.ipm.2020.102382.

Maglio, P. P., Srinivasan, S., Kreulen, J. T., & Spohrer, J. (2006). Service systems, service scientists, SSME, and innovation. *Communications of the ACM*, *49*(7), 81–85. http://dx.doi.org/10.1145/1139922.1139955.

Nowinski, W., & Kozma, M. (2017). How can blockchain technology disrupt existing business models? *Entrepreneurial Business and Economics Review*, *5*(3), 173–188. http://dx.doi.org/10.15678/EBER.2017.050309.

Oham, C., Michelin, R. A., Jurdak, R., Kanhere, S. S., & Jha, S. (2021). B-FERL: Blockchain based framework for securing smart vehicles. *Information Processing & Management*, *58*(1), Article 102426. http://dx.doi.org/10.1016/j.ipm.2020.102426.

Putz, B., Dietz, M., Empl, P., & Pernul, G. (2021). Ethertwin: Blockchain-based secure digital twin information management. *Information Processing & Management*, *58*(1), Article 102425. http://dx.doi.org/10.1016/j.ipm.2020.102425.

da Rosa Righi, R., Alberti, A. M., & Singh, M. (Eds.), (2020). *Blockchain Technology for Industry 4.0: Secure, Decentralized, Distributed and Trusted Industry Environment*. Singapore: Springer, ISBN: 978-981-15-1136-3, http://dx.doi.org/10.1007/978-981-15-1137-0.

Sharma, B., Thulasiram, R. K., Thulasiraman, P., Garg, S. K., & Buyya, R. (2012). Pricing cloud compute commodities: A novel financial economic model. In *CC GRID 2012* (pp. 451–457). Ottawa: IEEE, http://dx.doi.org/10.1109/CCGrid.2012.126.

Singh, A., Parizi, R. M., Zhang, Q., Choo, K.-K. R., & Dehghantanha, A. (2020). Blockchain smart contracts formalization: Approaches and challenges to address vulnerabilities. *Computers & Security*, *88*, Article 101654. http://dx.doi.org/10.1016/j.cose.2019.101654.

Tamburri, D. A., & Lago, P. (2011). Satisfying cloud computing requirements with agile service networks. In *2011 Services* (pp. 501–506). Washington, DC: IEEE, http://dx.doi.org/10.1109/SERVICES.2011.13.

Trestian, R., Ormond, O., & Muntean, G. (2012). Game theory-based network selection: Solutions and challenges. *IEEE Communications Surveys Tutorials*, *14*(4), 1212–1231. http://dx.doi.org/10.1109/SURV.2012.010912.00081.

Tsiropoulou, E. E., Katsinis, G. K., & Papavassiliou, S. (2012). Distributed uplink power control in multiservice wireless networks via a game theoretic approach with convex pricing. *IEEE Transactions on Parallel and Distributed Systems*, *23*(1), 61–68. http://dx.doi.org/10.1109/TPDS.2011.98.

Werner, S. M., Pritz, P. J., & Perez, D. (2020). Step on the gas? A better approach for recommending the Ethereum gas price. In *Springer Proceedings in Business and Economics*, *MARBLE 2020* (pp. 161–177). Vilamoura: Springer, http://dx.doi.org/10.1007/978-3-030-53356-4_10.

Wood, G. (2014). *Ethereum: A secure decentralised generalised transaction ledger*: *Yellow Paper Petersburg version 6424f7d: 2020-12-28*, (pp. 1–32). Parity.io, Online at: https://ethereum.github.io/yellowpaper/paper.pdf (Last Accessed: 5/Jan/21).

Xu, X., Sun, G., Luo, L., Cao, H., Yu, H., & Vasilakos, A. V. (2021). Latency performance modeling and analysis for hyperledger fabric blockchain network. *Information Processing & Management*, *58*(1), Article 102436. http://dx.doi.org/10.1016/j.ipm.2020.102436.

Zhao, Q., Chen, S., Liu, Z., Baker, T., & Zhang, Y. (2020). Blockchain-based privacy-preserving remote data integrity checking scheme for IoT information systems. *Information Processing & Management*, *57*(6), Article 102355. http://dx.doi.org/10.1016/j.ipm.2020.102355.

Zheng, Z., Xie, S., Dai, H.-N., Chen, W., Chen, X., Weng, J., et al. (2020). An overview on smart contracts: Challenges, advances and platforms. *Future Generation Computer Systems*, *105*, 475–491. http://dx.doi.org/10.1016/j.future.2019.12.019.