Blockchain Evaluation Approaches: State-of-the-Art and Future Perspective

sensors

MDPI

Review

Blockchain Evaluation Approaches: State-of-the-Art and Future Perspective

Sergey Smetanin [1], Aleksandr Ometov [2], Mikhail Komarov [1,*], Pavel Masek [3] and Yevgeni Koucheryavy [2]

1 National Research University Higher School of Economics, Moscow 101000, Russia; ssmetanin@hse.ru (S.S.); evgeny.kucheryavy@tuni.fi (Y.K.)
2 Tampere University, FI-33720 Tampere, Finland; aleksandr.ometov@tuni.fi
3 Brno University of Technology, 61600 Brno, Czech Republic; masekpavel@vutbr.cz
* Correspondence: mkomarov@hse.ru

Received: 28 April 2020; Accepted: 6 June 2020; Published: 13 June 2020

Abstract: The present increase of attention toward blockchain-based systems is currently reaching a tipping point with the corporate focus shifting from exploring the technology potential to creating Distributed Ledger Technology (DLT)-based systems. In light of a significant number of already existing blockchain applications driven by the Internet of Things (IoT) evolution, the developers are still facing a lack of tools and instruments for appropriate and efficient performance evaluation and behavior observation of different blockchain architectures. This paper aims at providing a systematic review of current blockchain evaluation approaches and at identifying the corresponding utilization challenges and limitations. First, we outline the main metrics related to the blockchain evaluation. Second, we propose the blockchain modeling and analysis classification based on the critical literature review. Third, we extend the review with publicly accessible industrial tools. Next, we analyze the selected results for each of the proposed classes and outline the corresponding limitations. Finally, we identify current challenges of the blockchain analysis from the system evaluation perspective, as well as provide future perspectives.

Keywords: blockchain; modeling; simulation; emulation; review

1. Introduction

Historically, blockchain systems were designed to support the data immutability among different decentralized nodes [1]. This niche of distributed systems development had already obtained a significant impact on business, touching upon all industries in the world as part of the Distributed Ledger Technology (DLT) paradigm [2]. Today's blockchain technology applications vary from public means of record-keeping and private storage [3] to connecting various heterogeneous devices as part of the Internet of Things (IoT) paradigm [4]. Moreover, IBM researchers forecast that the DLT market based on blockchain is expected to reach $60.7 billion by 2024 [5], highlighting the timeliness of the systems' integration needs. The tendency of the blockchain systems' adoption could be found in almost any digitalized industry, forcing companies to prepare to profit from the blockchain technology integration [6].

Deloitte's Global Blockchain Survey highlighted that the perception of the blockchain is presently reaching the point of no return, with the corporate focus shifting from exploring the technology potential to creating production business applications [7]. As one of the leading blockchain integration drivers, the financial sector accounted for more than 60% of the global blockchain market value in 2018 [8]. Since then, other industries have been cautious about finding use cases to ensure a good return of investment by implementing blockchain-based applications due to their extreme complexity

compared to conventional centralized systems. As a result, more and more entities in more and more spheres, for instance, the biological sciences, media, government agencies, communications, agriculture, and healthcare are broadening and developing their blockchain initiatives to keep up with the pace of the technological evolution [9,10].

Given the enormous interest in blockchain-based solutions, the number of startups is increasing yearly in both private and public sectors [11,12]. From the very beginning of the blockchain era, blockchain-related investments increased from about a million U.S. dollars in 2012 to more than one billion in 2017 [13], currently breaking the line of $23.7 billion [14]. A representative example is the blockchain protocol EOS, which has engendered a significant investment of $4.2 billion in its initial coin placement [15]. While the United States is presently considered a leader in the blockchain segment [16], it is also expected that China will be able to achieve this position in the coming years. One of the central reasons is that China's investments in blockchain technology are almost doubling each year [17].

A recent survey of business leaders in Europe [16] showed that almost 50% of them expected the blockchain to be added to their current operating business model. Moreover, another 33% claimed they expected the blockchain to be entirely replaced by their current operating model. About 66% of global companies expressed a moderate level of interest in blockchain technology, with almost 10% currently in the process of conducting experiments or implementing blockchain solutions [18].

Cryptocurrencies, and more specifically, Bitcoin, tend to be some of the first and central applications of the broadly known blockchain paradigm, receiving much more attention than other blockchain use cases. As of the third quarter of 2019, Bitcoin reached a high market capitalization of $205.4 billion U.S. dollars [19]. While Bitcoin could be considered the largest, some other cryptocurrencies such as Ethereum, Ripple, and Litecoin have also gained a significant market share in 2019 [20]. Many believe that blockchain and its use in cryptocurrency technology will allow shifting from traditional money transactions to digital ones supported by means of secure ledgers in the coming years.

As a result of the deep blockchain systems' penetration into our everyday lives, the number of distributed node interactions is expected to increase significantly [21], bringing more load to the energy grid [22], as well as infrastructure and peer-to-peer (P2P) networks and storage volumes [23]. While the field of conventional infrastructure-like (cloud) communication analysis and predictability is already a well-studied topic, the impact of new networking paradigms, such as fog and edge [24,25], is expected to bring a new level of system complexity both from communications and computing perspectives. Therefore, blockchains provide some unique differences from everything that has come before. A blockchain survives faults and attacks by the use of redundant checking of multiple nodes. This resiliency goes far beyond replication since it happens across the network without any central coordinator or intermediary [26]. Generally, the design of said distributed systems requires careful planning and performance evaluation, while conventional approaches may face numerous challenges due to the increase in the complexity.

To date, there is still a lack of unified tools and instruments for the performance evaluation and behavior observation of blockchains, while the number of related applications is already sky high [27]. Blind development of blockchain-based extraordinarily complex and dynamic systems without any preliminary performance evaluation may have a tremendous negative impact during the actual deployment phase. In most cases, current evaluation approaches are based on the emulation techniques that imitate and replicate the behavior of the entire network. Evidently, it requires a significant amount of computational, storage, and communications resources [28,29]. Little attention is also given to deploying real testnets driven by the community, but it requires significant incentivization activities to get the users involved [30,31]. Consequently, the emulation entails a massive problem of scalability for real-world deployments' evaluation [32]. Moreover, such an evaluation requires considerable engineering efforts to modify complicated open-source solutions or production systems to test out continually evolving systems in a timely manner. In this case, modeling approaches, for instance, analytical and simulation methods, can be considered as an alternative trading precision for the evaluation speed.

The main research question highlighted in this paper is:

> What are the currently used blockchain evaluation approaches and corresponding challenges?

This paper also aims at future prospects of blockchain simulation/modeling approaches, as well as at surveying existing solutions utilized for blockchain systems' performance evaluation. It aims to provide an overview of existing strategies found in the literature, as well as in open access sources in order to provide the linkage between the approach and solved task or its general area.

The rest of the paper is organized as follows. The next section provides the applied critical review methodology. Section 3 highlights the main background information, as well as the motivation of this paper. Section 4 outlines the examples of the most widely used approaches applicable to blockchain evaluation and related classification. Next, Section 5 covers the main analytical and simulation tools used for blockchain evaluation. Section 6 outlines the main emulation-based approaches used by industry and integrators. Next, Section 7 lists the main challenges related to the blockchain evaluation from both execution and legal perspectives. It also provides recommendations on the future improvement of the modeling process. The last section concludes the paper.

2. Methodology

In order to identify the key publications on the evaluation of blockchain technology, we performed a literature search in scientific databases following PRISMA guidelines [33] with minor modifications. The analysis covered leading computer science journals and conferences: IEEE Xplore, ACM Digital Library, ScienceDirect, SAGE Journals Online, Springer Link, etc. To find relevant articles and papers for our research, we applied the following search string: (Blockchain OR "Distributed Ledger") AND (Simulation OR Model OR Modeling OR Emulation OR Evaluation).

In total, we gathered a set of 1432 potentially relevant publications, excluding grey literature and pre-prints. We removed potential duplicates and arrived at 960 resources. We then analyzed the titles, keywords, and abstracts of the publications to identify papers and articles that described at least one modeling or simulation approach for blockchain-based systems. In doing so, we selected a total of 44 publications. To further extend our literature sample, we analyzed the references of the selected publications for additional papers or articles relevant to our research. Following this process resulted in a total of 63 publications. Finally, we analyzed publicly-available solutions proposed by the actual blockchain developers and arrived at a total of 71 publications.

Once the literature selection process was completed, we carefully read the selected sources to identify the described strategies and challenges. Next, we classified the extracted approaches into five general groups.

3. Background and Motivation

This section elaborates on the identified approaches presently used for the blockchain evaluation from analytical, simulation, and emulation perspectives. The majority of modern software systems rely on numerous practical and theoretical approaches for their performance evaluation. Nonetheless, Software Development Life Cycle Models (SDLC) fall short in providing the necessary flexibility for describing blockchain systems still keeping the priority of the initial performance evaluation before the actual system deployment, as well as followed by continuous reevaluation and testing [34]. Since the operation of the highly dynamic environment showed itself to be unstable multiple times in the past, the same should be applied to any pre-deployment phase of the blockchain systems.

Today, simulations and analytical modeling are standard instruments for the behavior and performance evaluation of the majority of blockchain-based solutions [35]. At the same time, analytical modeling could also be applied to the blockchain evaluation for cases when a mathematical model has a closed-form solution, i.e., a simplified description of the system operation should be given in the form

of a "set of equations" utilized to describe the behavior of the system formulated as a mathematical analytic function.

Simulation models could be either a pure but simplified simulation of the system behavior or, more often, could be considered as a subclass of mathematical models. In this case, the simulation would combine both mathematical and logical aspects of the system and try to replicate a real-life system behavior using computer software. Simulation tends to be used in cases when the analytical description cannot be formulated, or creating an analytical model is fundamentally impossible. Simulation attempts to approximate a system's behavior and development over time by running a model [36]. Those models are, in essence, generally simplified abstractions of a simulated system that aim at covering the specific level of details required to achieve the research goals. Simulations can demonstrate some limited effects of alternative conditions and action paths. This group of evaluation approaches is also used in cases where a real system cannot be engaged, for instance, when the system is not accessible, or it may be dangerous or unacceptable to access the system, or the system does not exist [37].

More broadly, computer simulation tries to approximate the system behavior and development over time by implementing and running a computer simulation model. By changing variables and conditions in the implemented simulation model, researchers can make predictions about the behavior of the simulated system without the need for the actual implementation of the entire system. A computer simulation is commonly used when it is a complicated task to accomplish the system emulation and for emulating the blockchain system as part of more complex environments [38].

According to [39], the simulation models can be classified along different dimensions: continuous or discrete, static or dynamic, deterministic or stochastic. A discrete model is a simulation model in which the state variables change only at a discrete set of points in time. A continuous model is a simulation model in which the state variables change continuously over time. A minimal amount of real-world systems is wholly continuous or discrete, but in the majority of cases, one type of change predominates over the other. A static system simulation model, which is also called Monte Carlo simulation, represents a system at a given point in time or represents one in which time does not matter. As the opposite of a static model, a dynamic simulation model represents a system with varying behavior over time. A model is called deterministic in the case that it does not include any random or probabilistic variables. Otherwise, it is called a stochastic simulation model.

Besides analysis and simulations, we can consider a standalone group of approaches used for the blockchain performance evaluation: the emulation of the entire or part of the system. It is the process of imitating the behavior that can be observed from the outside to match an existing target. The emulation mechanism's internal state does not need to reflect the internal state of the emulated system accurately. On the other hand, simulation involves modeling the internal state of the analyzed system, i.e., the output of a good simulation is that the simulation model will emulate the simulated system. Emulators, therefore, require significant overhead to run the emulated interface and/or system's functionality in real-time while providing the layers of virtualization needed to emulate a whole system. As a consequence, emulation seems to be more accurate in comparison with simulation but requires much computational resources to achieve it at the same time.

Currently, the field of basic modeling theory of blockchain systems is still in its infancy, including but not limited to constructing mathematical models, providing security and performance analysis, and identifying potential optimization points. In order to identify the publications primarily addressing blockchain modeling, we performed a search on a range of scientific databases that cover the high-quality computer science journals and conferences. Based on the literature review, we divided blockchain evaluation approaches into the following classes (see Figure 1): queuing models, Markov processes, Markov decision processes, random walks, and emulations. Some systems may have several types of models so that we can assume such a system as a mixed approach. The proposed classification is an extended one from [40]. The following subsections provide a more detailed overview of each technique.

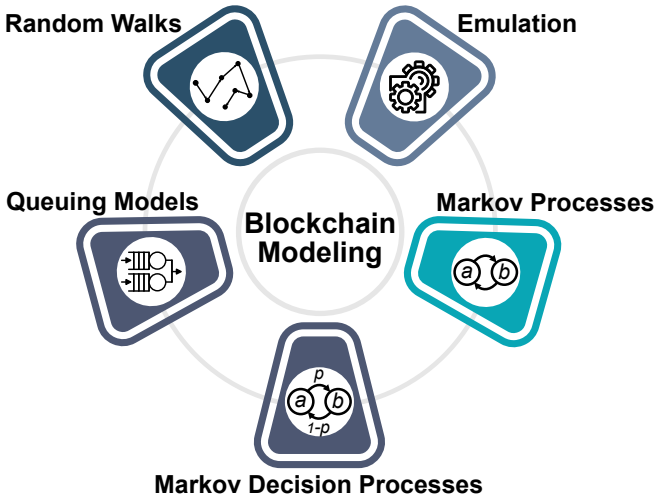

Figure 1. Proposed blockchain modeling approaches' classification.

4. Evaluation Strategies and Metrics

In order to better understand the reasons behind the need for blockchain evaluation, this section outlines the main metrics of interest and evaluation strategies applied in this process. To date, the academic literature lacks explicit analysis of the strategies due to the nascent field of research.

Generally, any blockchain-based system could be analyzed from various perspectives: usability, block analysis, functional testing, security analysis, integration, smart contracts, DApps, networks, and especially, performance evaluation, which is the main focus of this paper. Performance testing allows us to analyze the network size, and its ability to process transactions is critical, as it provides an opportunity to identify hardware and software bottlenecks in advance in addition to future Operational Expenses (OPEX). Performance-wise, the operation of a blockchain could be described by the set of the main metrics that could be classified into three main groups, as summarized in Table 1 (some metrics have multiple terms since the taxonomy is not yet defined by the community.): Blockchain metrics (the number of produced blocks, the number of processed transactions, processing time, finality time, etc.), P2P network metrics (the number of hit/miss requests, the number of active peers, the volume and structure of P2P traffic, etc.), and system node metrics (CPU, memory, storage, network, etc.).

Most of the metrics have the lower/average/upper bounds defined by the system configuration, network, or hardware limitations. Thus, developers should carefully consider those during the system design phase, while the numbers usually found in the project descriptions may be somewhat speculative, e.g., tps could not be measured in an objective way since it will require analysis on all the blockchain nodes, which is mostly unfeasible in a real-life environment. Similar problems could be found at lower levels, i.e., being related to the distributed P2P network operation. The heterogeneity of the communication environment, as well as unpredictable node distribution, may have an extensive impact on the metrics of interest, including block delivery and transaction validation times. Notable for enterprise (mainly private) blockchains is the specific definition of system metrics, e.g., a developer might measure transaction latency on public blockchains as when a transaction is available on 80% of nodes. However, for widely used enterprise consensus mechanisms, the threshold will often be set to 100%.

In truth, blockchain developers should consider the following metrics during the system design since the adjustments of already deployed distributed ones may be very challenging or even impossible.

Table 1. Main blockchain performance evaluation parameters (P) and metric (M).

	Metric	Type	Description	Main Challenge
Blockchain metrics	Consensus	P	The distributed process by which a network of nodes provides a guaranteed unique order of transactions and validates the block of transactions.	Inappropriate selection of a consensus algorithm may have a tremendous negative impact on the node and system operation.
	Transaction throughput	M	The rate at which valid transactions are committed by the blockchain network in a defined time period, commonly transactions per second (tps).	Careful selection of the system parameters is required in order to achieve the desired tps.
	Transaction type/size	P	The amount of data in the transaction to be added in the next block.	Inappropriate selection may have increased transaction fees in public blockchains.
	Block size	P	Defined size of the block, essentially the number of transactions to be included. It could be used to control the blockchain operation.	Larger blocks may have a negative impact on OPEX and decrease the tps.
	Chain size	M	The total size of the blockchain minus database indexes in megabytes.	A chain that is too long significantly decreases its distribution time to allow the new node to start the operation.
	Network-wide latency (commit time or transactional Latency)	M	The amount of time taken for a transaction to take effect to be used across the network.	Commit parameters may vary in different systems, e.g., 90% in public blockchains or 100% in enterprise private blockchains.
	Finality time	M	The moment when a transaction is committed and can no longer be reversed, i.e., the data cannot be rolled back to the previous state.	Defined in a consensus algorithm, and a threshold should be carefully selected during the evaluation. Otherwise, the finality time should significantly decrease the system efficiency.
Network metrics	Network size (number of active peers)	M/P	The number of validating nodes participating in consensus excluding observer nodes.	n/a
	Volume of P2P traffic	M	The amount of cumulative traffic generated by active nodes in the system.	Operation over public networks may have a tremendous negative impact on energy consumption, connection quality, as well as increased OPEX.
	Structure of underlying traffic	P	The structure of blockchain-related packets (data, service, etc.).	Inefficient selection may cause unnecessary traffic overheads.
	Packet loss ratio	M	The ratio between lost and sent packets related to blockchain operation.	Increased packet loss may increase delays and decrease tps.
Node metrics	CPU/GPU	M	Hardware utilized for blockchain-related data processing.	Has a significant impact on the involvement in the blockchain operation, as well as on the Capital Expenditures (CAPEX) and OPEX.
	Memory	M	The amount of RAM required for efficient transaction/block processing.	
	Storage volume	M	Local storage space required for the blockchain.	
	Connectivity	M	Various metrics of the selected communications technology, its channel quality, reliability, latency, etc.	
	Read latency	M	The time between the read request submission until the reply is received.	n/a
	Read throughput	M	A measure of how many read operations are completed in a defined time period expressed as reads per second (rps).	n/a
	Cache Hit Ratio (CHR)	M	Measurement of how many content requests a cache is able to fill successfully, compared to how many requests it receives.	Low CHR, mainly caused by hardware, may have an impact on the operational speed of a specific node.

5. Analytical and Simulation-Based Approaches

This section outlines the main academic activities related to the blockchain evaluation, while the industrial ones are given in Section 6. The summary of both groups is given in Table 2.

5.1. Queuing Models

A Queuing Model (QM) is a mathematical model allowing the prediction of the queue lengths and waiting time in the system. Some preliminary work on QM for a blockchain was carried out several

years ago by Kasahara and Kawahara [41]. This work studied the applications of queuing theory in the context of transaction-confirmation time for Bitcoin. The authors investigated and described the relationship between the demand for transactions with low fees and the transaction-confirmation time. Additionally, the authors demonstrated that the enhancement of the maximum block size appeared as an inefficient way to reduce the time of transaction confirmation. The development of a queuing analytical approach to the blockchain was continued in the work [42]. In both papers, the authors made the assumption that the transaction confirmation time follows a continuous probability distribution function. In order to define a system of differential-difference equations, the authors utilized the supplementary variable method by using the elapsed service time. Unfortunately, the authors did not provide the correctly defined solution of the formulated system. To overcome this issue, Li et al. introduced a generalized queuing theory of blockchain systems through the matrix-geometric solution in [43]. The proposed theory utilized different exponential service stages for times of blockchain-construction and block-generation.

Later on, Memon's group developed a new simulation model of the mining process for blockchain systems using queuing theory in [44]. The research group from Yale University proposed a stochastic model, which aimed at capturing the blockchain network dynamics and evolution [45]. Ricci et al. proposed a complicated framework composed of the machine learning model and queuing theory model [46]. It aimed at solving two significant tasks: identifying which transactions will be confirmed and characterizing the confirmation time of transactions.

Several recent studies utilized the fluid limit: a subsection in queuing theory describing deterministic processes, which aims at an approximation of the evolution of the analyzed stochastic process. For instance, the research group from the University of Amsterdam developed a Bitcoin-inspired infinite-server model with a random fluid limit [47]. Moreover, King et al. considered the fluid limit of a random graph model for a shared ledger and distributed ledger in blockchain systems [48].

5.2. Markov Processes

A Markov Process (MP) is a basic mathematical tool for performance evaluation of blockchain systems proposed more than 10 years ago in [49]. It describes a process by which future probabilities are determined by the most recent values. A stochastic process can be called an MP only in the case if:

$$P(X_{t_n} \lesssim X_n | X_{t_{n-1}}, \cdots, X_{t_1}) = P(X_{t_n} \lesssim X_n | Xt_{n-1}) \quad (1)$$

is true for every n and every $t_1 < t_2 < \cdots < t_n$ [50]. From the mathematical point of view, an MP is a first-order autoregressive model $X_t = c + aX_{t-1} + \epsilon_t$, where X_t is a time series, a is a parameter of the model, c is a constant, and ϵ_t is a white noise.

Some preliminary work was carried out by Eyal and Sirer in [51]. The authors constructed a basic Markov process for analyzing the vulnerability of blockchain protocols. They found out that selfish mining tends to become profitable in case the hashing power of a miner is larger than 25%. This approach was further developed by Nayak's research group by expanding the mining strategy with a novel "stubborn" strategy [52]. According to the results of the systematic exploration of strategy space, researchers found that the revenue of the attacker tends to increase by non-trivial combinations of network-level attacks and stubborn mining. In the paper [53], the authors explored the relationship between the existence of multiple misbehaving pools and the profitability of successful selfish mining.

The research group from Northeastern University developed a simple MP-based approach for analyzing the consistency properties of blockchain protocols [54]. While previous studies on consistency [55,56] argued that Markov models are too complicated for analysis, this research was the first one successfully using the MP-based models to analyze consistency against any adversary.

5.3. Markov Decision Processes

A Markov Decision Process (MDP) is a discrete time stochastic process, which is widely used as a mathematical framework for a sequential decision-making task with a fully observable environment described by a Markov transition model and additional rewards. MDP can be defined as a four-tuple (S, A, P_a, R_a), where S is the finite set of states, A is the finite set of actions, $P_a(s, s') = Probability(s_{t+1} = a | s_t = s, a_t = a)$ is the probability that action a in state s moves to state s' at state s_{t+1}, and $R_a(s, s')$ is the expected reward received immediately after transitioning from state s to state s' by performing action a.

In order to identify an optimal mining policy in blockchain systems, several studies utilized MDP as a mathematical modeling framework. For instance, several studies [57–59] utilized MDP to find the optimal selfish mining strategy. Bitcoin Simulator [58] was initially implemented to examine how network characteristics, modification of the consensus protocol, and consequence parameters affect the efficiency, security, and scalability of Proof-of-Work (PoW) powered blockchain systems and is also partially based on MDP. The framework consists of two main components: a blockchain simulator and a blockchain security model utilizing the MDP approach. In order to make the simulator as realistic as possible, the authors incorporated real blockchain networks' statistics, for instance, the number of nodes, the geographical distribution of nodes, block size distribution, time of block generation, network delays, and techniques for information propagation. The primary output of the blockchain simulator component is the stale block rate, which is then processed by the blockchain security model. The output of the security model, which is based on MDP, enables researchers to identify optimal adversarial strategies by comparing the performance and security of blockchain systems with different parameters. Based on the comparison of the security aspects of Bitcoin and Ethereum, the authors found that in order to match Bitcoin's security with six block confirmations, Ethereum needed at least 37 block confirmations. Bitcoin Simulator is implemented based on the widely-known discrete-event Network Simulator 3 [60], which has a scalability limit of 6000 nodes. Bitcoin networks currently have more than 10,000 nodes as of May 2020 [61], so technically, Bitcoin Simulator is not able to perform the simulation of the current state of the entire Bitcoin network [62].

5.4. Random Walks

A Random Walk (RW) is a stochastic mathematical model that describes a path that consists of random changes or steps on mathematical space at discrete points in time, widely utilized for various Information and Communications Technology (ICT) systems' analysis [63,64]. Given independent and identically distributed random variables $X_1, X_2, \ldots, X_n$, where $X_i \in R_n$, the pure structure of random walks can be defined as $Yn = \sum_{1}^{n} X_i$.

The authors of [65] refined Nakamoto's model [23] to study the double-spending attack issue in blockchains, primarily focusing on the likelihood of attack success. Jang and Lee [66] proposed a new model, which takes into account block confirmation release in contrast with Goffard's model. Grunspan and Perez-Marco, within their research [66,67], computed the minimal number of confirmations to be requested by the recipient such that the double-spend strategy was non-profitable.

To summarize, most of the blockchain evaluation approaches based on analytical modeling and simulations are generally simplified abstractions of the actual systems aiming at a specific focus on particular aspects of the system operation, thus providing highly limited observations on the main blockchain evaluation metrics listed in Table 2. Therefore, those are generally found in academic works in contrast to another broad field of the system emulation driven by actual integrators.

Table 2. State-of-the-art of the blockchain evaluation analytical and simulation approaches.

#	Field of Study	Str.	Specifics
[41,42]	*Bitcoin:* Transaction-confirmation time	QM	The authors made the assumption that the times of the transaction confirmation follow a continuous probability distribution function.
[43]	*Bitcoin:* The queuing theory of blockchain systems under a dynamic behavior setting.	QM	Analysis of the average mean of transactions in the Memory Pool (MemPool), mean number of transactions in a block, and mean transaction-confirmation time. It is noted that considering more general blockchain queuing systems is required in the future.
[44]	*Bitcoin:* Behavior of blockchain-based applications before deploying them over the blockchain network	QM	The authors assumed that the number of arrivals to the system was a little more than those serviced at the mining station; however, in Bitcoin systems, the number of accumulated unconfirmed transactions can be observed to increase throughout the day.
[45]	*Ethereum:* valuation of selfish mining strategies' impact.	QM	The work was focused on measurements, blockchain design, and analysis. It was stated that careful examination of the various system parameters' relationships and related trade-offs should be a base for similar models' development.
[46]	*Bitcoin:* Profitability of double spending attacks and delays.	QM	The authors assumed no off-chain payment mechanism with no simplified payment verification during the transaction confirmation. They stated that QMs were the most suitable for the analysis of transaction delays.
[47]	*Bitcoin:* The fluid limit of a random graph model for a shared ledger.	QM	It was stated that a more realistic FIFO-batch departure discipline is required in order to analyze a natural $G/M/\infty$-like Bitcoin queue variant of the $G/M/\infty$-like Bitcoin.
[48]	*IOTA:* The fluid limit of a random graph model for a shared ledger.	QM	The author attempted to analyze IOTA scholastically and find the differential equation for the delays in the system, i.e., a converged fluid limit.
[51]	*Bitcoin:* Incentive-compatible property of Blockchain systems.	MP	The authors highlighted the importance of the chain-quality property since if the adversary controls a suitably limited amount of hashing power, then the adversary is also limited in terms of the number of blocks it has added.
[52]	*Bitcoin:* Selfish mining attack.	MP	The authors stressed that a malicious user in this system would eventually succeed in case he/she executed double-spending simultaneously, thus defining a new research direction of stubborn mining.
[53]	*Bitcoin:* Selfish mining attack.	MP	The work observed the relation of the number of malicious selfish miners vs. the selected one success rate.
[54]	*GHOST(Ethereum):* Consistency properties of blockchain protocols.	MP	The work proposed a model that considered various attacks and aspects of the protocols' operation. The authors assumed that some random genesis blocks were available from the initial trusted setup. They also stated that Markov models needed to be extremely complex in order to capture the dynamics of such complicated systems.
[57]	*Bitcoin:* Selfish mining attack.	MDP	Profit threshold for selfish mining attacks aimed at capturing different blockchain instances, which have various stale block rates and confirmations. The authors stressed that a malicious user in this system would eventually succeed in case he/she executed double-spending simultaneously.
[58]	*Bitcoin, Litecoin, and Dogecoin:* Selfish mining attack.	MDP	The authors proposed a flexible model with flexible integration of different PoW blockchain instances. It was supported by a standalone security abstraction layer. The systems allowed executing various attacks, corresponding optimal execution strategies, and the impact on the operation.
[59]	*Bitcoin:* Selfish mining attack defense strategy.	MDP	The authors analyzed the current fork resolution issue and highlighted the need for its revision by introducing the fork-resolving policy based on weights.
[65]	*Bitcoin:* Double spending attack.	RW	The author proposed to analyze probabilistically the executability of the attack. The implementing block confirmation mechanism was considered as a significant potential improvement [67].
[66]	*Bitcoin:* Double spending attack.	RW	The authors proposed the analysis of the different mining strategies' profitabilities, in particular with the honest strategy. It was mentioned that difficulty adjustment during the attack should be considered in more detail when similar techniques are developed.

6. Emulation-Based Approaches

Emulation (E) is the process of imitating the behavior that can be observed from the outside to match the selected target, which is mainly used by the actual blockchain developers having access to the actual system code. Most of the industry and integrators apply this approach to test the performance of their private or consortium blockchain-based solutions, while public ones obtain much less attention. The details on the main blockchain emulation examples are summarized in Table 3. The reasoning behind this is that the first two groups originate from highly directional business processes driven by corporations that follow conventional SDLC strategies in order to decrease CAPEX and OPEX costs. Generally, blockchain emulators are expected to provide an overview of the actual system operation, therefore providing a more in-depth analysis of the system metrics from Table 1.

To start with, an industrial project titled Hyperledger [68] is one of the blockchain-adoption drivers known to the world today. Started in 2015 and driven by both the community as well as Linux Foundation, it has already received contributions from IBM, Intel, and SAP Ariba. The project has resulted in a set of DLT tools, including Hyperledger Composer and Caliper, aiming to support Hyperledger Fabric pre-deployment activities, unfortunately lacking many features of the actual blockchain operation [69]. The main idea behind this tool development is to allow easy and on-the-fly testing of smart contractsin a variety of simplified system operation abstractions, thus accelerating the actual deployment process. It consists of a data model, a set of transactions, and a set of queries by which those transactions can access data within the model, thus allowing for the execution of various microservices in the emulator. A number of academic activities were presented with respect to this framework [70].

Another emulation tool is Blockbench [28], which is an academia-driven framework for stress testing and analyzing private blockchains. It measures component-wise and overall system performance in terms of scalability, throughput, latency, and fault-tolerance. The authors conducted experiments on Hyperledger Fabric, Parity, and Ethereum, showing that these blockchains were still far from replacing current database systems in the case of traditional data processing workloads. The authors claimed that any private blockchains could be emulated or used nodes could be instantiated in the network.

Next, VIBES [71], a public blockchain emulator, was initially designed to correct the deficiencies in the currently stopped Bitcoin-Simulator project [72]. In order to achieve this goal, VIBES upgrades the scalability characteristics of the Bitcoin-Simulator by spanning multiple threads to carry out expensive computations and achieve full CPU utilization. As Bitcoin Simulator, VIBES utilizes the same method of inserting timestamps to execute the PoW hypothetically. Both simulators appear to be trailblazers to simulate the entire blockchain network. Moreover, in contrast with previous simulators, VIBES is designed to simulate transactions in the blockchain network. As an input, VIBES processes the network parameters, as well as the blockchain system-wide characteristics. Previous input parameters joined with theoretical and empirical results are used to simulate the blockchain system with a vast number of nodes. Based on these data, VIBES does not require performing heavy computations so that it can speed up the simulation process. However, the system scalability in the experiments is kept below 1500 nodes.

BlockLite [62] is another tool designed to emulate the public blockchain operation on a single node with both high usability and scalability. In contrast to VIBES, BlockLite is comprised of a module to execute real PoW workload, scales up to 20,000 nodes, and operates as a fully decentralized system. In order to deal with the scalability challenge of the real PoW loads, BlockLite delegates one node to solve a puzzle in a pre-running stage and then replicates the behavior of the delegation node in a running application.

A particular niche of blockchain evaluation is related to private blockchains. Here, the developer may either develop his/her completely proprietary blockchain solution or base it on one of the open systems adopted for private use. One of the examples of the last group is Ethereum [73]. Many developers base their private blockchain systems on Ethereum, varying from conventional finance to industrial applications [74], by means of an enterprise solution. Customer-oriented projects

require initial performance evaluation, and thus, Ethereum Tester was specifically designed for the deep emulation of said systems [75]. Besides observing the specific metrics stated, Ethereum Tester allows analyzing the operation of the Application Programming Interface (API), web integration, back-end, smart contracts, and several other blockchain tests.

Another toolbox utilized for private Ethereum-based blockchains' evaluation is known as Truffle Suite [76]. It is composed of a set of tools, including blockchain emulation, smart contacts, and transaction tracing in a virtual machine. The system is composed of three main components covering the main operational aspects [77]. The first component, called Truffle, is the actual development environment that integrates the compilation, testing, and deployment of smart contracts. Ganache is the second one. It is a locally deployed blockchain simulator featuring a graphical user interface that can simulate blockchain networks and live-test smart contracts without requiring setting up real test networks or using a remote network. The last one is Drizzle, being an assortment of front-end libraries that offer useful components for developing web applications that can seamlessly connect with the smart contracts. The system is extremely flexible, but dedicated only to Ethereum analysis.

A massive open-source environment called Corda Testing Tools provides a broad range of functionality, including unit tests and integration tests for both small projects and enterprises from the transaction perspective designed explicitly for R3 Corda project analysis [78]. The main idea behind the project is to analyze the potential of interoperability between different parties with personal blockchain systems. The performance of Corda significantly varies based on the hardware and system complexity, e.g., from 8 to 1001 tps according to the official documentation [79].

One more toolbox for private blockchain integration testing is Exonum Testkit [80]. Its main targets are testing the logic of transactions and analysis of custom APIs in the Exonum blockchain, as well as the lack of consensus algorithms' evaluation. Exonum Testkit is written in the Rust programming language, but also provides a Java Binding tool as a part of its software development kit (SDK).

It must be noted that there are some minor projects developed to analyze a particular part of the blockchain operation, but not for the actual blockchain evaluation, thus falling out of the scope of this paper. For example, Manticore [81] is a tool developed to analyze the behavior of Ethereum smart contracts from the dynamic symbolic execution perspective [82]. This field may be of interest for virtual environment developers. A similar analysis at the low level was done in [83,84]. Some other projects are dedicated to the Distributed Application (DApp) deployment strategies in Ethereum as Truffle Drizzle [85]. Numerous other tools are dedicated to the standalone node emulation in Ethereum, such as Geth [86] or EthereumJS Monorepo [87]. BitcoinJ is a similar project, but utilized for Bitcoin. It is a Java-based implementation of the Bitcoin protocol for the emulation of transactions and wallet operation [88]. It has a number of abstractions allowing users to analyze the simplified payment verification, asynchronicity, and per-connection status. The main drawback of those is the limitation in evaluating the overall system operation.

To summarize, one of the critical aspects of constructing emulation models tends for the academic segment to be the computational resource limitations. Broadly utilized solutions for this issue are to simplify resource-competitive modules, utilize multi-threading programming in model implementation, and to deploy emulation models in the cloud. The impact of resource limitations for product developers and integrators is less critical; thus, most of their activities are targeted at the emulation environment. It allows users to obtain more detailed information on the system operation. Evidently, a unified solution for blockchain emulation does not exist, while this section listed the main aspects of existing projects in this field, providing the reader with an opportunity to select the tool of interest depending on the application from Table 3.

Table 3. Main blockchain emulation tools: PU, Public Blockchain; PR, Private Blockchain.

#	Tool	Type	Scalability	Main Application	Main Challenge
[28]	Blockbench	PU/PR	Low	First academia-driven framework allowing analyzing different blockchains, e.g, Ethereum, Parity, and Hyperledger Fabric.	The project is inactive.
[62]	BlockLite	PU	Low	An academia-driven work aimed to allow the emulation of the blockchain on a single node.	The project is inactive.
[79]	Corda Testing Tools	PR	High	Enterprise solution allowing evaluating Corda operation, as well as new DApps before the actual integration.	Limited to R3 Corda.
[74]	Ethereum Tester	PR	n/a	Ethereum-driven open-source library allowing testing DApps and smart contracts.	Designed specifically for Ethereum developers and lacks community support.
[80]	Exonum Testkit	PR	High	Private (enterprise) Exonum blockchain emulator.	Limited to Exonum.
[69]	Hyperledger Tools	PR	High	Community-driven private blockchain performance analysis toolbox supporting high tps and various implementations.	Extreme complexity of the development environment.
[76]	Truffle Suite	PR	High	Allows flexible development of Ethereum-based DApps and smart contracts' operation with strong community support.	n/a
[71]	VIBES	PU	Medium	An academia-driven emulator developed for Bitcoin system analysis. An excellent educational example with detailed explanations of the system operation.	The project is inactive.

During the review, we identified that major analytical models are based on Markov processes, which are commonly utilized for examining the incentive-compatible property of blockchain systems, studying selfish mining attack strategies, and evaluating the consistency properties of blockchain protocols. At the same time, models based on Markov decision process are commonly used for obtaining relatively better results on examining selfish mining attack strategies. The majority of papers aiming at researching double spending attacks in blockchain systems are based on the random walks approach. Emulation approaches should be applied in cases when the detailed performance evaluation is required.

7. Current Challenges and Future Prospects

The models listed in Section 3 primarily focus on merely imitating and replicating the behavior of blockchain networks, while reducing the number of computational resources in contrast with the original blockchain implementations. Based on the analyzed literature, we identified two groups of challenges and prospects explicitly related to blockchain evaluation rather than to general blockchain operation; see Figure 2.

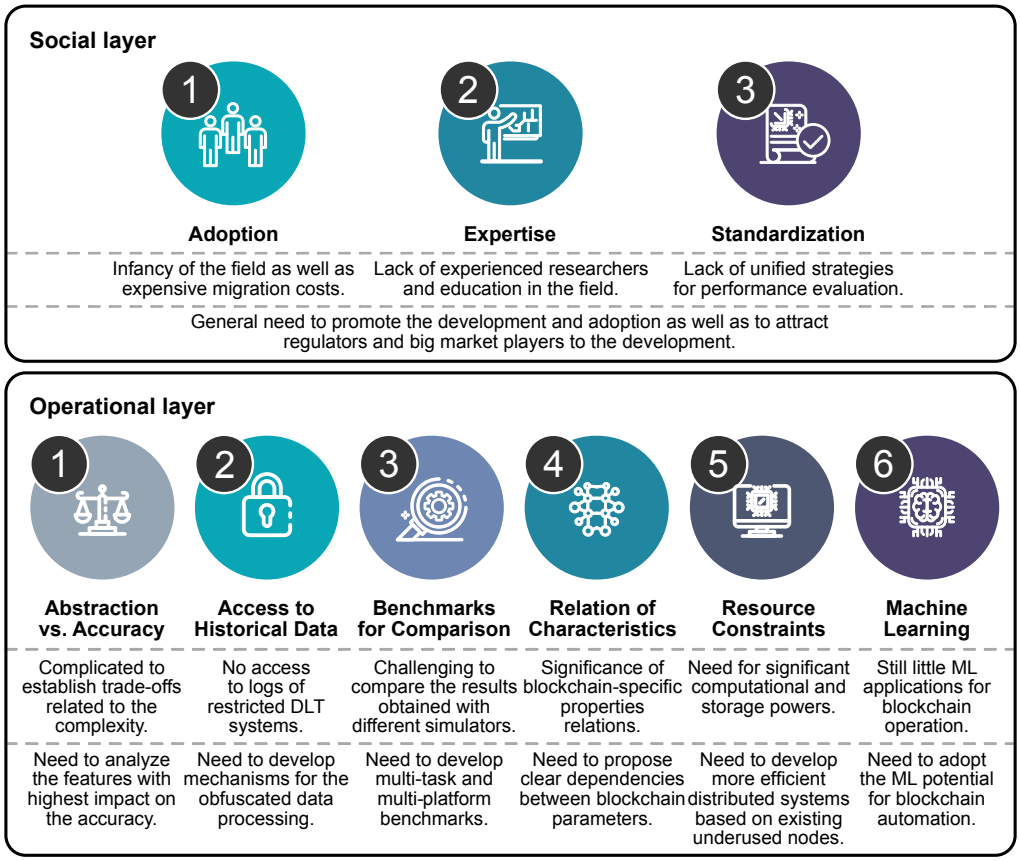

Figure 2. Challenges and perspectives of blockchain simulation systems.

7.1. Lack of Adoption

The first and the main social challenge is related to limited adoption of the technology. Adding to developers' bad luck, many affiliate blockchains with cryptocurrency used for money laundering and tax fraud [89]. It has not publicly received much support from the government due to its historical opposition to control [90]. Besides, the mass adoption of the technology remains very limited due to numerous factors, e.g., institutional, environmental, technological and many others [91]. In contrast, the top management of companies already shows interest in deploying blockchain-based systems, according to [92], which is expected to result in additional activities related to the developed systems' evaluation needs.

7.2. Lack of Expertise

One of the main challenges of today's blockchain system developers is a lack of skills or experience in analyzing and developing blockchain applications. The steps from centralized to massively adopted distributed systems are only to be made in the coming years. Learning additional skills or understanding the best practices to implement blockchain applications is costly. It is essential to manifest the fact that blockchain technology has enormous potential for simplifying all the bureaucratic procedures for the state and making transactions transparent and more accessible, thus pushing the need for the systems' evaluation to a higher priority.

7.3. Lack of Standardization

Based on the literature review, we can state that the evaluation of blockchain systems highly varies depending on the type, i.e., private, public, or consortium [93]. Generally, public blockchains are analyzed by targeting some specific research goals, while private and consortium blockchains are

driven by business needs (reducing CAPEX/OPEX, system planning, peak performance evaluation, etc.). This is one of the results due to which most of the high precision evaluation tools are provided by private blockchain developers and lie in the emulation class. Moreover, the audit or analysis of private blockchain systems may sometimes be even impossible due to internal company regulations or the General Data Protection Regulation (GDPR); it may be beneficial to define a common standardized approach for private blockchain evaluation in order to reach consistency while comparing different enterprise products.

7.4. A Multi-Task Benchmark for the Models' Comparison

Even if several articles focused on one problem [52,53,57–59], it may be challenging to compare the outcomes in a straightforward way since even a slight difference in the test data may affect the final results significantly. Thus, one of the significant current challenges can be formulated as the lack of a multi-task benchmark and analysis platform for performance and accuracy evaluation of simulated models. In order to overcome this issue, new experimental studies should rely on the experience of developing such a multi-task benchmark and analysis platform for other computer science fields of study. For instance, in computational linguistics, GLUE [94] was developed for analyzing and evaluating the performance metrics of models across a wide variety of existing natural language understanding tasks.

7.5. Access to the Representative Historical Data of Blockchain Systems

Following the previous challenge, historical data such as logs collected from monitoring nodes or general network statistics are commonly analyzed and utilized in simulation models. However, this approach suffers from several drawbacks. For instance, in the case of obtaining data from specific nodes, these data are under the analyzed blockchain system factors. As a consequence, it may not be representative to examine the blockchain system, which is under other network factors. Besides, these data are observed from logs of a limited amount of nodes, so they are not able to describe other nodes' behavior [95]. Moreover, in both cases, it can be a challenging task to collect these data from private networks. Blockchain developers and blockchain system owners can solve this challenge by publishing representative statistics from different levels of blockchain networks.

7.6. Abstractions Used in a Simulation Model May Affect the Accuracy of the Model

Event-based simulation models, e.g., [71,96], perform the simulation by abstracting part of the node logic of the whole node into a range of discrete events triggered at scheduled periods of time. Even if these abstractions increase scalability and cost-effectiveness, they may ultimately result in the exclusion of the essential traits of blockchain systems due to the abstraction of the nodes' functionalities [95]. One of the possible, but rather complicated solutions is to validate empirically the influence of abstracted functionalities on the blockchain system by conducting comparison tests.

7.7. Relations Between Blockchain Characteristics

While certain relations or even trade-offs between quantitative blockchain characteristics were deeply examined, e.g., the effect of the block-size limit to the transaction-confirmation time [41,42], the relations between more complicated characteristics have remained open. For instance, several blockchain characteristics are challenging to assess, e.g., cost, law, and regulation. As a consequence, in order to simulate systems that take into account these characteristics, more detailed and testable criteria must be generated. Some outstanding work in identifying trade-offs between distributed ledger technology characteristics was conducted in the paper [97], so it can be used as a starting point for further analysis of this field.

7.8. Resource Constraints

In the case of modeling blockchain systems, running a simulation or an emulation may require significant computational resources in order to achieve relevant and accurate results. The basic approaches for reducing the need for computational power are to simplify resource-consumptive modules or procedures in the model implementation, or to utilize a certain central processing unit's abilities such as multithreading, or to exclude some modules from the model. More advanced and combined approaches may be required to run the execution of a large-scale blockchain model on tens of thousands of nodes in the cloud [62].

7.9. Machine Learning in Simulation Modeling

While Machine Learning (ML) solutions demonstrated an ability to perform successfully in system modeling tasks [98–101], a minimal amount of studies applied those for the modeling of blockchain-based systems. For instance, the Markov chain neural networks concept introduced by Awiszus and Rosenhahn [102] has great potential in the application of Markov chains for blockchain modeling. At the same time, the Markov Decision Process Extraction Network (MPEN) [103] can potentially be utilized in order to extract automatically minimal relevant aspects of the dynamics from observations to model a Markov decision process.

Based on the above, we can conclude that there exists a number of challenges to be addressed by the research and professional community pushed by the growing industrial and academic interest in blockchain modeling. Several issues, e.g., the development of a benchmark, the estimation of the influence of model abstractions, and the relations between blockchain characteristics, can be solved by researchers on their own, whereas some other challenges, e.g., access to representative historical data, could be overcome only by the whole blockchain-development community participating. At the same time, a few potential directions, such as machine learning techniques and cloud-based deployment, seem to be able to enforce the simulation accuracy and performance notably.

8. Conclusions

In this paper, we executed a critical review, analyzed the state-of-the-art of blockchain evaluation approaches, and identified current challenges and future prospects of blockchain simulation and modeling.

Firstly, we outlined the main motivation and background. Next, we listed the main perspectives and metrics that could be evaluated. Further on, we proposed a classification of blockchain modeling approaches into the following classes: Queuing Models, Markov Processes, Markov Decision processes, random walks, and emulations. We executed the literature review based on the PRISMA methodology extended by the industrial projects and reviewed selected papers for each of the proposed classes. Finally, we outlined current challenges and future perspectives in the area of blockchain evaluation.

Based on the critical review, we concluded that analytical and simulation approaches based on queuing theory are mainly utilized for evaluating different blockchain architectures before deploying them over the blockchain network and designing the fluid limit of a random graph model for a shared ledger. Approaches based on Markov processes are commonly used for evaluating the consistency properties of blockchain protocols and studying various attack strategies. As an extension of the Markov processes, approaches based on Markov decision processes are commonly applied for obtaining relatively better results on examining selfish mining attacks. Approaches based on random walks are primarily used to examine double spending attacks in blockchain systems. Emulation is used when the computational and storage resources are not critical in contrast to previous ones. Those are mainly applied by actual developers and integrators when detailed system analysis is required for planning of private and enterprise blockchains.

Finally, we identified current challenges and future prospects of blockchain simulation approaches, which include a lack of expertise, the need for developing a multi-task benchmark for reliable models'

comparison, gaining access to the representative historical data of blockchain systems, evaluating the influence of abstractions on the model accuracy, exploring relations between blockchain characteristics, reducing computational resources for simulation, and the usage of machine learning for better performance. We foresee that one of the main bottlenecks of blockchain evaluation adoption is the need for standardization activities, which is expected to be resolved in the oncoming years.

Author Contributions: Conceptualization, S.S. and A.O.; methodology, A.O.; validation, M.K., P.M., and Y.K.; formal analysis, P.M.; investigation, S.S. and A.O.; writing, original draft preparation, S.S.; writing, review and editing, A.O.; visualization, A.O.; supervision, A.O.; project administration, M.K., and Y.K.; funding acquisition, M.K. All authors have read and agreed to the published version of the manuscript.

Funding: The research was financed by the Technology Agency of the Czech Republic (TACR) under Grant No. TK02030013. For the research, infrastructure of the SIX Center was used.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zheng, Z.; Xie, S.; Dai, H.; Chen, X.; Wang, H. An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. In Proceedings of the IEEE International Congress on Big Data (BigData congress), Honolulu, HI, USA, 25–30 June 2017; pp. 557–564.
2. Bellini, E.; Iraqi, Y.; Damiani, E. Blockchain-based Distributed Trust and Reputation Management Systems: A Survey. *IEEE Access* **2020**, *8*, 21127–21151.
3. Sunyaev, A. Distributed Ledger Technology. In *Internet Computing*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 265–299.
4. Khan, M.A.; Salah, K. IoT Security: Review, Blockchain Solutions, and Open Challenges. *Future Gener. Comput. Syst.* **2018**, *82*, 395–411.
5. WinterGreenResearc, Inc. *Blockchain Market Shares, Market Strategies, and Market Forecasts, 2018 to 2024*; Industry Survey; Lexington, MA, USA, 2018.
6. Pawczuk, L.; Massev, R.; Schatsky, D. *Breaking Blockchain: Open Deloitte's 2018 Global Blockchain Survey*; Deloitte Consulting: Denver, CO, USA, 2018.
7. Insights, D. Deloitte's 2019 Global Blockchain Survey. In *Blockchain Gets Down to Business*; Deloitte Consulting: Denver, CO, USA, 2019.
8. IDC. Distribution of Blockchain Market Value Worldwide in 2018. 2018. Available online: https://www.idc.com/getdoc.jsp?containerId=prAP43559618 (accessed on 8 June 2020).
9. Researcht, B. *Global Blockchain in Agriculture and Food Market: Focus on Stakeholders, Regulations, Application (Supply Chain Tracking, Finance Management, Data Management, and Land and Property Ownership) and Regional Adoption*; BIS Research: Fremont, CA, USA, 2018.
10. PwC. Blockchain Survey. 2018. Available online: https://www.pwc.com/blockchainsurvey (accessed on 8 June 2020).
11. Dimitrov, B. What Changed? Enterprise Blockchain Startups Are All of A Sudden Cool. 2020. Available online: https://www.forbes.com/sites/biserdimitrov/2020/02/19/what-changed-enterprise-blockchain-startups-are-all-of-a-sudden-cool/ (accessed on 8 June 2020).
12. Bizinger, D. EU Startups: Blockchain. 2020. Available online: https://www.eu-startups.com/tag/blockchain/ (accessed on 8 June 2020).
13. CBI Insights. *Blockchain in Review: Investment Trends And Opportunities*; Technical Report; CB Information Services, Inc.: New York, NY, USA, 2019.
14. Lawrence, L.; Joel, J. Investments in Blockchain 2019: $23.7 Billion Raised by 3738 Blockchain Companies Since 2013. 2019. Available online: https://outlierventures.io/research/investments-in-blockchains-2019-23-7-billion-raised-by-3738-blockchain-companies-since-2013/ (accessed on 8 June 2020).
15. CoinDesk. ICO Tracker. 2018. Available online: https://www.coindesk.com/ico-tracker/ (accessed on 8 June 2020).
16. Blechschmidt, B.B. Blockchain in Europe: Closing the Strategy Gap. *Cognizant* **2018**. Available online: https://www.cognizant.com/whitepapers/blockchain-in-europe-closing-the-strategy-gap-codex3320.pdf (accessed on 8 June 2020).

17. Blockchain White Paper—China Academy of Information and Communication Technology. 2018. Available online: http://www.caict.ac.cn/english/yjcg/bps/201901/P020190131402018699770.pdf (accessed on 8 June 2020).
18. Gartner. Organizations' Blockchain Plans. 2018. Available online: https://www.gartner.com/newsroom/id/3873790 (accessed on 8 June 2020).
19. Blockchain Market Cap. 2019. Available online: https://www.blockchain.com/charts/market-cap (accessed on 8 June 2020).
20. TradingView. Cryptocurrency Market. 2019. Available online: https://www.tradingview.com/markets/cryptocurrencies/global-charts/ (accessed on 8 June 2020).
21. Alphand, O.; Amoretti, M.; Claeys, T.; Dall'Asta, S.; Duda, A.; Ferrari, G.; Rousseau, F.; Tourancheau, B.; Veltri, L.; Zanichelli, F. IoTChain: A Blockchain Security Architecture for the Internet of Things. InProceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 15–18 April 2018; pp. 1–6.
22. De Vries, A. Bitcoin's Growing Energy Problem. *Joule* **2018**, *2*, 801–805.
23. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. White Papper; Manubot: Panama City, Panama, 2019.
24. Busanelli, S.; Cirani, S.; Melegari, L.; Picone, M.; Rosa, M.; Veltri, L. A Sidecar Object for the Optimized Communication Between Edge and Cloud in Internet of Things Applications. *Future Internet* **2019**, *11*, 145.
25. Mäkitalo, N.; Ometov, A.; Kannisto, J.; Andreev, S.; Koucheryavy, Y.; Mikkonen, T. Safe, Secure Executions at the Network Edge: Coordinating Cloud, Edge, and Fog Computing. *IEEE Softw.* **2017**, *35*, 30–37.
26. Hyperledger Blockchain Performance Metrics: White Paper. 2018. Available online: https://www.hyperledger.org/wp-content/uploads/2018/10/HL_Whitepaper_Metrics_PDF_V1.01.pdf (accessed on 8 June 2020).
27. Lao, L.; Li, Z.; Hou, S.; Xiao, B.; Guo, S.; Yang, Y. A Survey of IoT Applications in Blockchain Systems: Architecture, Consensus, and Traffic Modeling. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 1–32.
28. Dinh, T.; Anh, T.; Wang, J.; Chen, G.; Liu, R.; Ooi, C.; Tan, K.L. Blockbench: A Framework for Analyzing Private Blockchains. In Proceedings of the 2017 ACM International Conference on Management of Data, Chicago, IL USA, 14–19 May 2017; pp. 1085–1100.
29. Chen, C.; Qi, Z.; Liu, Y.; Lei, K. Using Virtualization for Blockchain Testing. In *International Conference on Smart Computing and Communication*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 289–299.
30. Zhidanov, K.; Bezzateev, S.; Afanasyeva, A.; Sayfullin, M.; Vanurin, S.; Bardinova, Y.; Ometov, A. Blockchain Technology for Smartphones and Constrained IoT Devices: A Future Perspective and Implementation. In Proceedings of the 21st Conference on Business Informatics (CBI), Moscow, Russia, 15–17 July 2019; Volume 2, pp. 20–27.
31. A. Ometov, et al. An Overview on Blockchain for Smartphones: State-of-the-Art, Consensus, Implementation, Challenges and Future Trends. *IEEE Access [Early Access]* **2020**. doi:10.1109/ACCESS.2020.2998951.
32. Faria, C.; Correia, M. BlockSim: Blockchain Simulator. In Proceedings of the IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 439–446.
33. Moher, D.; Liberati, A.; Tetzlaff, J.; Altman, D.G. Preferred Reporting Items for Systematic Reviews and Meta-analyses: The PRISMA Statement. *Ann. Intern. Med.* **2009**, *151*, 264–269.
34. Miraz, M.H.; Ali, M. Blockchain Enabled Smart Contract Based Applications: Deficiencies with the Software Development Life Cycle Models. *arXiv* **2020**, arXiv:2001.10589.
35. Memon, R.A.; Li, J.P.; Ahmed, J. Simulation model for blockchain systems using queuing theory. *Electronics* **2019**, *8*, 234.
36. Banks, J. *Discrete Event System Simulation*; Pearson Education: London, UK, 2005.
37. Sokolowski, J.A.; Banks, C.M. *Principles of Modeling and Simulation: A Multidisciplinary Approach*; John Wiley & Sons: Hoboken, NJ, USA, 2011.
38. Pirmagomedov, R.; Ometov. A.; Moltchanov, D.; Lu, X.; Kovalchukov, R.; Olshannikova, E.; Andreev, S.; Koucheryavy, Y.; Dohler. M. Applying Blockchain Technology for User Incentivization in mmWave-based Mesh Networks. *IEEE Access* **2020**, *8*, 50983–50994.
39. Law, A.; Kelton, D. *Simulation Modeling and Analysis*; McGraw-Hill Education: Frisco, TX, USA, 2000; Volume 5.

40. Li, Q.L.; Ma, J.Y.; Chang, Y.X.; Ma, F.Q.; Yu, H.B. Markov Processes in Blockchain Systems. *Comput. Soc. Netw.* **2019**, *6*, 1–28.
41. Kawase, Y.; Kasahara, S. Transaction-confirmation Time for Bitcoin: A Queueing Analytical Approach to Blockchain Mechanism. In Proceedings of the International Conference on Queueing Theory and Network Applications, Qinhuangdao, China, 21–23 August 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 75–88.
42. Kasahara, S.; Kawahara, J. Effect of Bitcoin Fee on Transaction-confirmation Process. *J. Ind. Manag. Optim.* **2019**, *15*, 365–386.
43. Li, Q.L.; Ma, J.Y.; Chang, Y.X. Blockchain Queue Theory. In Proceedings of the International Conference on Computational Social Networks; Springer: Berlin/Heidelberg, Germany, 2018; pp. 25–40.
44. Memon, R.A.; Li, J.; Ahmed, J.; Khan, A.; Nazir, M.I.; Mangrio, M.I. Modeling of Blockchain Based Systems Using Queuing Theory Simulation. In Proceedings of the 15th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 14–16 December 2018; pp. 107–111.
45. Papadis, N.; Borst, S.; Walid, A.; Grissa, M.; Tassiulas, L. Stochastic Models and Wide-Area Network Measurements for Blockchain Design and Analysis. In Proceedings of the IEEE Conference on Computer Communications (INFOCOM), Honolulu, HI, USA, 16–19 April 2018; pp. 2546–2554.
46. Ricci, S.; Ferreira, E.; Menasche, D.S.; Ziviani, A.; Souza, J.E.; Vieira, A.B. Learning Blockchain Delays: A Queueing Theory Approach. *ACM SIGMETRICS Perform. Eval. Rev.* **2019**, *46*, 122–125.
47. Frolkova, M.; Mandjes, M. A Bitcoin-inspired Infinite-server Model with a Random Fluid Limit. *Stoch. Models* **2019**, *35*, 1–32.
48. King, C. The Fluid Limit of a Random Graph Model for a Shared Ledger. *arXiv* **2019**, arXiv:1902.05050.
49. Bolch, G.; Greiner, S.; De Meer, H.; Trivedi, K.S. *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2006.
50. Bharucha-Reid, A.T. *Elements of the Theory of Markov Processes and Their Applications*; Courier Corporation: New York, NY, USA, 2010.
51. Eyal, I.; Sirer, E.G. Majority is Not Enough: Bitcoin Mining is Vulnerable. In Proceedings of the International Conference on Financial Cryptography and Data Security, Christ Church, Barbados, 3–7 March 2014; Springer: Berlin/Heidelberg, Germany, 2014, pp. 436–454.
52. Nayak, K.; Kumar, S.; Miller, A.; Shi, E. Stubborn Mining: Generalizing Selfish Mining and Combining with an Eclipse Attack. In Proceedings of the IEEE European Symposium on Security and Privacy (EuroS P), Saarbrucken, Germany, 21–24 March 2016; pp. 305–320. doi:10.1109/EuroSP.2016.32.
53. Bai, Q.; Zhou, X.; Wang, X.; Xu, Y.; Wang, X.; Kong, Q. A Deep Dive into Blockchain Selfish Mining. In Proceedings of the IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019; pp. 1–6.
54. Kiffer, L.; Rajaraman, R.; Shelat, A. A Better Method to Analyze Blockchain Consistency. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 729–744.
55. Garay, J.; Kiayias, A.; Leonardos, N. The Bitcoin Backbone Protocol with Chains of Variable Difficulty. In Proceedings of the Annual International Cryptology Conference, Santa Barbara, CA, USA, 20–24 August 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 291–323.
56. Pass, R.; Seeman, L.; Shelat, A. Analysis of the Blockchain Protocol in Asynchronous Networks. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, 30 April–4 May 2017; Springer: Berlin/Heidelberg, Germany, 2017, pp. 643–673.
57. Sapirshtein, A.; Sompolinsky, Y.; Zohar, A. Optimal Selfish Mining Strategies in Bitcoin. In Proceedings of the International Conference on Financial Cryptography and Data Security, Christ Church, Barbados, 22–26 February 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 515–532.
58. Gervais, A.; Karame, G.O.; Wüst, K.; Glykantzis, V.; Ritzdorf, H.; Capkun, S. On the Security and Performance of Proof of Work Blockchains. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 3–16.
59. Zhang, R.; Preneel, B. Publish or Perish: A Backward-compatible Defense Against Selfish Mining in Bitcoin. In Proceedings of the Cryptographers' Track at the RSA Conference, San Francisco, CA, USA, 14–17 February 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 277–292.

60. NS-3. Network Simulator. 2019. Available online: https://www.nsnam.org/ (accessed on 8 June 2018).
61. Bitnodes. Global Bitcoin Nodes Distribution. 2020. Available online: https://bitnodes.earn.com/ (accessed on 8 June 2018).
62. Wang, X.; Al-Mamun, A.; Yan, F.; Zhao, D. Toward Accurate and Efficient Emulation of Public Blockchains in the Cloud. In Proceedings of the International Conference on Cloud Computing, June San Diego, CA, USA, 25–30 June 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 67–82.
63. Orlov, Y.; Kirina-Lilinskaya, E.; Samuylov, A.; Ometov, A.; Moltchanov, D.; Gaimamaka, Y.; Andreev, S.; Samouylov, K. *Time-Dependent SIR Analysis in Shopping Malls Using Fractal-Based Mobility Models*; Springer: Cham, Switzerland, 2017; pp. 16–25.
64. Spitzer, F. *Principles of Random Walk*; Springer: New York, NY, USA 2013; Volume 34.
65. Goffard, P.O. Fraud Risk Assessment within Blockchain Transactions. *Adv. Appl. Probab.* **2019**, *51*, 443–467.
66. Grunspan, C.; Pérez-Marco, R. On Profitability of Nakamoto double spend. *arXiv* **2019**, arXiv:1912.06412.
67. Jang, J.; Lee, H.N. Profitable Double-Spending Attacks. *arXiv* **2019**, arXiv:1903.01711.
68. Dhillon, V.; Metcalf, D.; Hooper, M. The Hyperledger Project. In *Blockchain Enabled Applications*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 139–149.
69. Welcome to Hyperledger Composer GitHub Repository. 2020. Available online: https://hyperledger.github.io/composer/v0.19/introduction/introduction.html (accessed on 8 June 2018).
70. Thakkar, P.; Nathan, S.; Viswanathan, B. Performance Benchmarking and Optimizing Hyperledger Fabric Blockchain Platform. In Proceedings of the 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Milwaukee, WI, USA, 25–28 September 2018; pp. 264–276.
71. Stoykov, L.; Zhang, K.; Jacobsen, H.A. VIBES: Fast Blockchain Simulations for Large-Scale Peer-to-Peer Networks. In Proceedings of the ACM/IFIP/USENIX Middleware Conference: Posters and Demos, Las Vegas, NV, USA, 11–15 December 2017; pp. 19–20.
72. Bitcoin-Simulator, Capable of Simulating Any Re-parametrization of Bitcoin. 2016. Available online: https://github.com/arthurgervais/Bitcoin-Simulator (accessed on 8 June 2018).
73. Pongnumkul, S.; Siripanpornchana, C.; Thajchayapong, S. Performance Analysis of Private Blockchain Platforms in Varying Workloads. In Proceedings of the 26th International Conference on Computer Communication and Networks (ICCCN), Vancouver, BC, Canada, 31 July–3 August 2017; pp. 1–6.
74. Ethereum for Enterprise. 2020. Available online: https://ethereum.org/enterprise/ (accessed on 8 June 2018).
75. Ethereum Tester GitHub Repository. 2020 Available online: https://github.com/ethereum/eth-tester (accessed on 8 June 2018).
76. Truffle: Smart Contracts made Sweeter. 2020. Available online: https://www.trufflesuite.com/truffle (accessed on 8 June 2018).
77. Sahu, M. What is Truffle Suite? Features, How to Install, How to Run Smart Contracts. 2020. Available online: https://www.upgrad.com/blog/what-is-truffle-suite/ (accessed on 8 June 2018).
78. Brown, R.G. *The Corda Platform: An Introduction*; White Paper; Corda: Dublin, Ireland, 2018.
79. R3 Ltd. Corda Enterprise 4.2: Sizing and performance. 2020. Available online: https://docs.corda.net/docs/corda-enterprise/4.2/sizing-and-performance.html (accessed on 8 June 2018).
80. Exonum Documentation. 2020. Available online: https://exonum.com/doc/version/latest/ (accessed on 8 June 2018).
81. Manticore GitHub Repository. 2020. Available online: https://github.com/trailofbits/manticore (accessed on 8 June 2018).
82. Mossberg, M.; Manzano, F.; Hennenfent, E.; Groce, A.; Grieco, G.; Feist, J.; Brunson, T.; Dinaburg, A. Manticore: A User-friendly Symbolic Execution Framework for Binaries and Smart Contracts. In Proceedings of the 34th IEEE/ACM International Conference on Automated Software Engineering (ASE), San Diego, CA, USA, 11–15 November 2019; pp. 1186–1189.
83. Permenev, A.; Dimitrov, D.; Tsankov, P.; Drachsler-Cohen, D.; Vechev, M. VerX: Safety Verification of Smart Contracts. In Proceedings of the IEEE Symposium on Security and Privacy, Genoa, Italy, 18–20 May 2020; pp. 18–20.

84. Hildenbrandt, E.; Saxena, M.; Zhu, X.; Rodrigues, N.; Daian, P.; Guth, D.; Rosu, G. KEVM: A Complete Semantics of the Ethereum Virtual Machine. In Proceedings of the IEEE 31st Computer Security Foundations Symposium (CSF), Oxford, UK, 9–12 July 2018; pp. 204–217.
85. Mohanty, D. Frameworks: Truffle and Embark. In *Ethereum for Architects and Developers*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 181–195.
86. Geth GitHub Repository. 2019. Available online: https://github.com/ethereum/go-ethereum/wiki/geth (accessed on 8 June 2018).
87. EthereumJS Monorepo GitHub Repository. 2020. Available online: https://github.com/ethereumjs/ethereumjs-vm (accessed on 8 June 2018).
88. bitcoinj GitHub Repository. 2020. Available online: https://github.com/bitcoinj/bitcoinj (accessed on 8 June 2018).
89. Houben, R.; Snyers, A. Cryptocurrencies and Blockchain: Legal Context and Implications for Financial Crime, Money Laundering and Tax Evasion. In *Technical Report, IPOL | Policy Department for Economic, Scientific and Quality of Life Policies*; European Parlaiment: Brussels, Belgium, 2018.
90. Maurer, B.; Nelms, T.C.; Swartz, L. "When Perhaps the Real Problem is Money Itself!": The Practical Materiality of Bitcoin. *Soc. Semiot.* **2013**, *23*, 261–277.
91. Janssen, M.; Weerakkody, V.; Ismagilova, E.; Sivarajah, U.; Irani, Z. A Framework for Analysing Blockchain Technology Adoption: Integrating Institutional, Market and Technical Factors. *Int. J. Inf. Manag.* **2020**, *50*, 302–309.
92. Clohessy, T.; Acton, T. Investigating the influence of organizational factors onblockchain adoption: An innovation theory perspective. *Ind. Manag. Data Syst.* **2019**, *119(7)*, pp. 1457–1491. DOI:10.1108/IMDS-08-2018-0365.
93. Akram, S.V.; Malik, P.K.; Singh, R.; Anita, G.; Tanwar, S. Adoption of Blockchain Technology in Various Realms: Opportunities and Challenges. *Secur. Priv.* **2020**, e109, doi:10.1002/spy2.109 .
94. Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In Proceedings of the 7th International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
95. Alsahan, L.; Lasla, N.; Abdallah, M. Local Bitcoin Network Simulator for Performance Evaluation Using Lightweight Virtualization. *arXiv* **2020**, arXiv:2002.01243.
96. Aoki, Y.; Otsuki, K.; Kaneko, T.; Banno, R.; Shudo, K. SimBlock: A Blockchain Network Simulator. In Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Moscow, Russia, 23–27 September 2019; pp. 325–329.
97. Kannengießer, N.; Lins, S.; Dehling, T.; Sunyaev, A. Mind the Gap: Trade-offs between Distributed Ledger Technology Characteristics. *arXiv* **2019**, arXiv:1906.00861.
98. Recknagel, F. Applications of Machine Learning to Ecological Modelling. *Ecol. Model.* **2001**, *146*, 303–310.
99. Tariq, H.; Al-Sahaf, H.; Welch, I. Modelling and Prediction of Resource Utilization of Hadoop Clusters: A Machine Learning Approach. In Proceedings of the 12th IEEE/ACM International Conference on Utility and Cloud Computing, Auckland, New Zealand, 2–5 December 2019; pp. 93–100.
100. Dueben, P.D.; Bauer, P. Challenges and Design Choices for Global Weather and Climate Models Based on Machine Learning. *Geosci. Model Dev.* **2018**, *11*, 3999–4009.
101. Mosavi, A.; Salimi, M.; Faizollahzadeh Ardabili, S.; Rabczuk, T.; Shamshirband, S.; Varkonyi-Koczy, A.R. State of the Art of Machine Learning Models in Energy Systems, a Systematic Review. *Energies* **2019**, *12*, 1301.
102. Awiszus, M.; Rosenhahn, B. Markov Chain Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2180–2187.
103. Duell, S.; Hans, A.; Udluft, S. The Markov Decision Process Extraction Network. In Proceedings of the European Symposium On Artificial Neural Networks, Computational Intelligence and Machine Learning, Haifa, Israel, 21–24 June 2010.