CLIENT PERFORMANCE PREDICTIONS FOR PRIVATE BLOCKCHAIN NETWORKS

László Viktor Jánoky[1], János Levendovszky[2] and Péter Ekler[1]

[1]Department of Automation and Applied Informatics, Budapest University of Technology and Economics, Budapest, Hungary
[2]Department of Networked Systems and Services, Budapest University of Technology and Economics, Budapest, Hungary

ABSTRACT

The recent public adaptation of cryptocurrencies sparked a great interest in alternative uses of the blockchain technology. Private or permissioned blockchain-based systems are a promising technology, initiating novel applications in several important fields, such as financing, commerce, and administration. One of the largest challenges in its application is the necessity of capacity planning. In public blockchains – such as the ones powering cryptocurrencies – the network is self-scaling and self-organizing, made up of individual nodes working for profit. In private blockchain, where capacity is provided by a few selected parties, these abilities are not inherently present as there is no financial or other motivation for clients to participate. This necessitates the introduction of efficient capacity planning and performance predictions to operate such a network efficiently. In this paper, we deal with methods for providing performance predictions of private blockchains.

KEYWORDS

Blockchain, Private blockchain networks, Performance prediction, Linear predictors

1. INTRODUCTION

Blockchain technology provides data storage in an unmodifiable, undeniable way. These facts led to the proliferation of cryptocurrencies and the rise of interest in alternative uses of the technology.

A key factor in achieving these desirable properties is the presence of a self-organizing network of clients. In a public Blockchain, the most common application of the technology, these clients work toward a common goal, motivated by their direct economic interest. Each client can read and write to the chain and their changes are validated by each participant until a consensus is reached. This is the working model behind cryptocurrencies, such as Bitcoin [1], but it is not the only use of the technology.

A subclass of Blockchains are private or permissioned blockchains (PBCs, sometimes also referred to as distributed ledgers [2]), where operations on the chain are limited to a certain subset of clients. This allows for much wider application of the technology while bringing in new challenges and problems to solve, especially regarding performance and efficiency. After a brief literature review, these problems will be more precisely stated and formalized in Section 3 (Problem Statement).

DOI: 10.5121/ijcnc.2020.12507

The rest of the paper is organized as follows. Section 4 deals with possible approaches, that are discussed and formally introduced. Section 5 is dedicated to the validation of these approaches by applying them in a practical case and drawing conclusions from the observations. Finally, Section 6 wraps up the discussion by providing a short overview of the work done and introducing possible future directions.

2. LITERATURE REVIEW

In this section, we summarize the main influential works in the field of blockchain technology. As the research on private blockchain technologies is a relatively recent field, the number of significant papers is relatively few.

After the initial use, blockchain technology has found many other applications than cryptocurrencies. However, the original whitepaper by the author with the pseudonym Satoshi Nakamoto (which describes the workings of Bitcoin) [1] is still considered the most important publication in the field. This paper introduces the basic concept of a Blockchain and also provides a use for the technology in cryptocurrencies.

While adequate for the use in Bitcoin, the original concept of Blockchain, using a Proof-of-Work [3] scheme for appending new blocks, has some performance issues which cannot be easily solved without sacrificing security. This was recognized relatively early and several studies addressed this topic. One of the most cited ones is titled as *On the Security and Performance of Proof of Work Blockchains* by A Gervais et al [4]. In this work, the authors introduced a framework for analyzing the performance and security implications of different consensus protocols in different implementations.

With the increasing adaptation of the technology, the performance issues became more apparent, even in the case of public blockchains. This led to a search for other alternatives with even more emphasis on performance analysis. One notable work is *The Quest for Scalable Blockchain Fabric: Proof-of-Work vs. BFT Replication* by Marko Vukolic [5], which deals with the performance and scalability issues of these style of systems.

While the focus of the literature is on the performance and scalability problems of large, public blockchains, a relatively few publications deal explicitly with the specifics of private blockchains. There are individual performance comparisons of popular private blockchain platforms, such as Ethereum [6] and Hyperledger Fabric [7], described in the paper *Performance analysis of private blockchain platforms in varying workloads* by S Pongnumkul, et al [8]. There are also initiatives at unifying the performance and security evaluation of private blockchains, such as the BLOCK BENCH solution, described in the paper [9] by Tien Tuan Anh Dins et al.

Another possible source of inspiration comes from related fields, such as network security. Ad-hoc networks built from untrusting parties face very similar problems as blockchains do. Proposed solutions in this field, such as the work of A. A. Pirzada et al on *Establishing Trust in Pure Ad-hoc Networks* [10] or the work of P. Chatterjee on *Trust Based Clustering and Secure Routing Scheme for Mobile Ad Hoc Networks* [11], provide some valuable insight to these challenges.

3. PROBLEM STATEMENT

PBCs share some of the well-known challenges associated with distributed systems, especially blockchain-based solutions, but also have some unique problems. In this section, we list these specific problems.

3.1. The Unique Case of PBCs

Typically, in PBCs the number of nodes is lower by several orders of magnitude than in case of public chains. This can be attributed to different use-cases and the restrictions on participants. To put it simply; there is no motivation for an outside party to use a private chain which they cannot write. On the other hand, an organization which can write data in the chain, usually lack the necessary computational resources to run the chain effectively, while maintaining data integrity. Consequently, the standard performance governing methods - such as block difficulty target in Proof-of-Work schemes [3] - which rely on the statistical behavior of a large number of nodes may prove to be unsatisfactory in this case. The approach described in this paper aims to provide more fine-grained control over performance for smaller PBCs, enabling their faster adaptation.

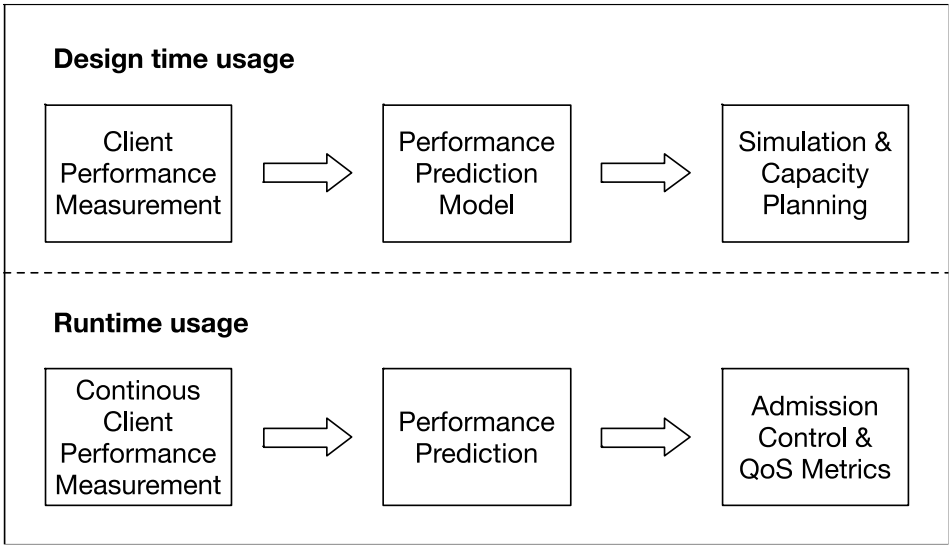

Figure 1. Uses of performance prediction in real world scenarios

By incorporating performance prediction both in design time and in runtime (as seen in Figure 1), further improvements can be achieved. The accurate modeling of client performance aids the capacity planning process of a new PBC, and continuous runtime measurements and predictions could help with giving more precise QoS guarantees.

3.2. Formal Description

To accurately state the problem, one must first describe the inner workings of the chain and its environment in a formal way. Only then can the problem of performance predictions for private blockchains be addressed.

A blockchain, as its name implies is made up from different data blocks, chained together by cryptographic links. Each block contains a certain amount of data (which is specific to that implementation), which is usually a fixed number.

In the most popular scheme, a node appends a new block to the chain when a cryptographic riddle, the so-called Proof-of-Work [3] is solved. On average, solving the Proof-of-Work takes the same amount of computational power regardless the content of the block, thus this cost can be considered a constant denoted by h for hardness (expressed in computational capacity units). In practice the cryptographic riddle usually entails the execution of several hash functions, ultimately leading to the discovery of an acceptable block. This approach was first introduced as part of the HashCash algorithm [12], which aimed to prohibit DDOS attacks but gained popularity with its implementation in Blockchain technologies.

The "total solving capacity of the chain", denoted as u (*capacity/sec*) is the total available computational power available, which can be tasked to solve proof-of-work tasks. Because u is dependent on the number of nodes in the system at any given time, it can be considered as a function of time, denoted by $u(k)$, where k represents the observed time step.

To determine the capacity of the chain (the block creation rate denoted r_b), it is enough to know the hardness h and the available computational power $u(k)$ at any given time. As previously discussed, hardness is either constant or it is defined by a predetermined algorithm known to all nodes. This means that the task can be reduced to the prediction of available computational power in a given time.

For constructing a formal model, let each node j belong to a class l. This class denotes the node's typical computational characteristics (i.e. dedicated server, or a mobile device). The process describing the offered capacity by the node j of class l has memory and its time dependence is denoted by $X_j^{(l)}(k)$.

4. PROPOSED CLIENT PERFORMANCE PREDICTION SOLUTION

As was shown in Section 3.2., a key factor to increase the efficiency and performance of PBCs is to have an accurate prediction of their performance in case of relatively small (as opposed to public chains) number of nodes.

We have examined several solutions to the problem stated above: (i) the first approach was to use a purely statistical approach described in [13], (ii) after that, we laid out the basic principles of the method discussed in [14]. It was shown that by using predictions a more accurate estimate can be given on the available computational resources, thus system performance itself can be more predictable in a statistical sense.

4.1. Using Linear Predictors

The key concept behind the solution is to provide dynamically changing predictions based on the observed measures, instead of static predictions based on historical trends. The first step in doing that is to provide an algorithm, which predicts the performance available in the next time-step based on the previous values.

Due to the time dependence and memory of the process defining the provided computational power by node j , one can use e.g. a linear predictor to predict the future value of $X_j^{(l)}(k)$ based on its past values

$$X_j^{(l)}(k-1), X_j^{(l)}(k-2), \ldots, X_j^{(l)}(k-V) \tag{1}$$

in the form of

$$\widetilde{X}_j^{(l)}(k) = \sum_{u=1}^{V} w^{(lj)} X_j^{(l)}(k-u) \tag{2}$$

With the predictor implemented, the offered computational capacity of a node can be modeled as a time series in the following way:

$$X_j^{(l)}(k) = \sum_{u=1}^{V} w_u^{(lj)} X_j^{(l)}(k-u) + e^{(l,j)}(k) \tag{3}$$

where

$$w_{opt}^{(l,j)}: \min_{w^{(l,j)}} E\left(X_j^{(l)}(k) - \sum_{u=1}^{V} w^{(lj)} X_j^{(l)}(k-u) \right)^2 \tag{4}$$

which entails that

$$E\left(e^{(l,j)2}\right) \leq E\left(X_j^{(l)2}\right) \tag{5}$$

due to the fact, that the predictor minimized the mean square error. With the capacity predicted for each node, the total available capacity can be easily calculated by summarizing the values:

$$Y(k) = \sum_{i=1}^{L} \sum_{j=1}^{n_j^{(1)}+\ldots+n_j^{(M)}} X_j^{(i)}(k) \tag{6}$$

There are several algorithms to find the optimal weight vector, in our case the most easily implementable one is the Robbins-Monroe type of stochastic approximations [15]. The steps of the algorithm are summarized as follows:

1. An initial weight $w_u^{(l,j)}(0)$ is chosen for a node j of class l. These initial values could either be based on historical measurements of the class or chosen as a vector taking all previous values with equal weights.
2. In each k time step (when new data is available) the weights are updated using the Robbins-Monroe formula:

$$w_u^{(l,j)}(k+1) = w_u^{(l,j)}(k+1) - \Delta\left\{ X_j^{(l)}(k) - \sum_{v=1}^{V} W_v^{(l,j)} X_j^{(l)}(k-v) \right\} X_j^{(l)}(k-u), u = 1, \ldots, V$$

3. Finally, a prediction is made using the linear predictor.

In typical real-world scenarios, individual node level measurements and performance predictions are usually inefficient or unfeasible. In those cases, the algorithm can be used with a larger granularity, predicting on client class level. This, of course, comes at the cost of accuracy but can be done with minimal modifications to the algorithm. One can use this method by omitting class notations l and replacing each client j with their respective class.

5. VALIDATION

To validate the approach, we built an environment which simulates the real-world implementation of a typical private blockchain. The simulation has aimed at providing client performance predictions, thus factors like network traffic, node failures were simplified. The next step in our work would be a real-world implementation of a PBC, using our algorithm, which would provide more detailed data and feedback based on the measurements.

5.1. Simulation Setup

In the setup, we measure client performance by running a synthetic benchmark on several different clients at different times. The performance was measured by how many SHA-256 Hash [16] operations the client can execute in a one-second time interval. The results of these benchmarks were then saved as a time-series, which can be found in the appendix.

With the data gathering complete over several measurement sessions of simulated clients, the results were cleaned, merged and run through the simulation framework. During the simulation, previously collected time-series were replayed, as if they were happening in real-time, which were used to feed data to the predictor. Based on the incoming data, our algorithm made a prediction, which could be immediately validated against the actual data measured earlier. To evaluate the measurement, we used two different metrics:

- As the basis for comparison between the approaches, we used the Root Mean Square Error (RMSE) of the predicted and actual values.
- Client performance prediction is evaluated with the analysis of the error distribution.
    - A lower expected value indicates a more precise prediction.
    - A low variance of the error distribution also signals good prediction.

5.2. Numerical Results

The results presented in this section were achieved using data from 20 client performance measurements where the clients belonged to 2 different classes. Each client was measured for a duration of 100-time steps, thus the total data points come to 2000. The unit of measurement is the previously discussed SHA256 Hash operation/seconds. Table 1 shows the main characteristics of the measured data.

Table 1. Main characteristics of the measured data.

Data points	Min	Max	Average	Standard Deviation
2000	40016	23639	37750,93	1736,71

The prediction results are compared to a baseline, non-predictive approach described in our previous paper [13]. The baseline is based on a simple statistical method, which operates only on the assumed distribution of the client without prediction.

As previously described in Section 3.2, in most cases individual client level prediction is unfeasible. For a more realistic scenario, the prediction algorithm also works on client class level, albeit with worse accuracy. In our measurements, we tested both approaches, the results of client level prediction can be seen in Table 2, while Figure 2 and Figure 3 describes the error distributions of the different methods respectively.

Table 2. Results of client level prediction.

Metric	**Baseline**	**Proposed Method**
Root Mean Square Error	1850	792
Standard Deviation of the Error Distribution	1300	1136

Figure 2. The baseline distribution of prediction errors in client level measurements

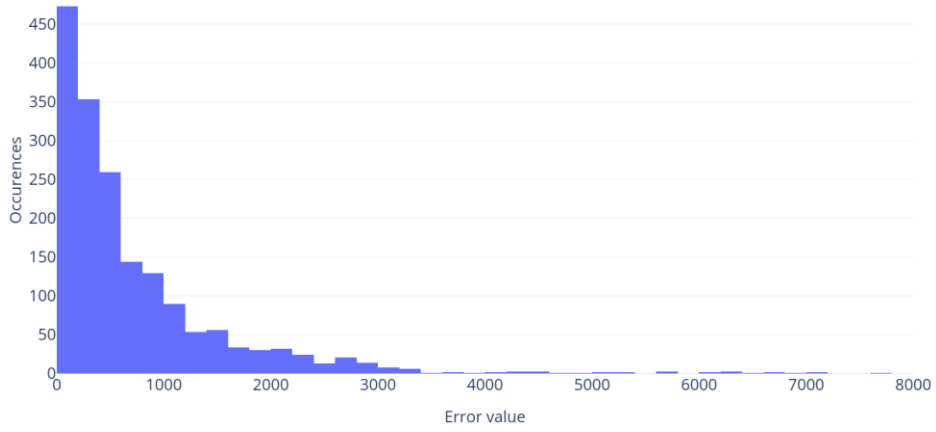

Figure 3. The distribution of prediction errors in case of client level measurements with the proposed method

In the case of client level prediction, the improvement provided by the proposed method is visible. However, in real-world scenarios, the additional costs narrow the applications of this approach. In the case of client level predictions, the actual measurements could outweigh the computational power of the clients themselves, thus group-level predictions become more desirable. Table 3, Figures 4 and 5 shows how group level prediction compares to the client level.

Table 3. Results of group-level prediction.

Metric	**Baseline**	**Proposed Method**
Root Mean Square Error	27032	4424
Standard Deviation of the Error Distribution	7755	3484

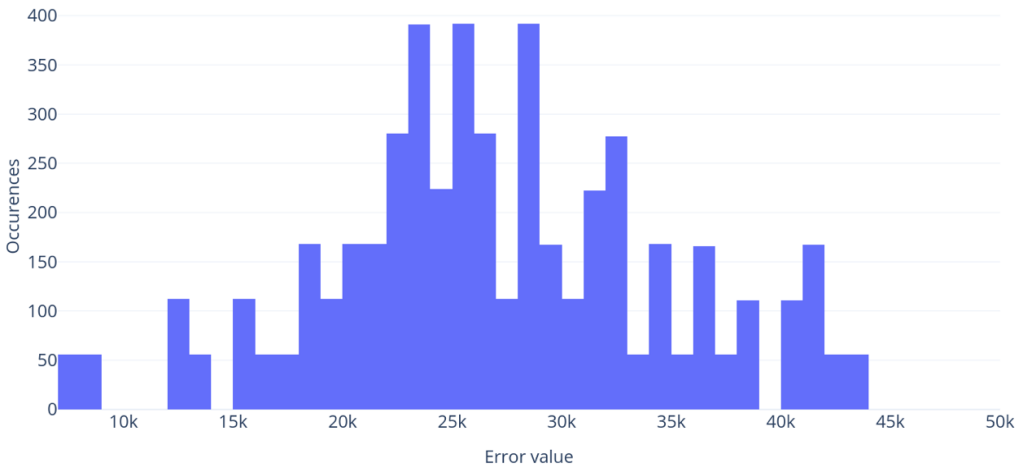

Figure 4. The baseline distribution of prediction errors in group level measurements

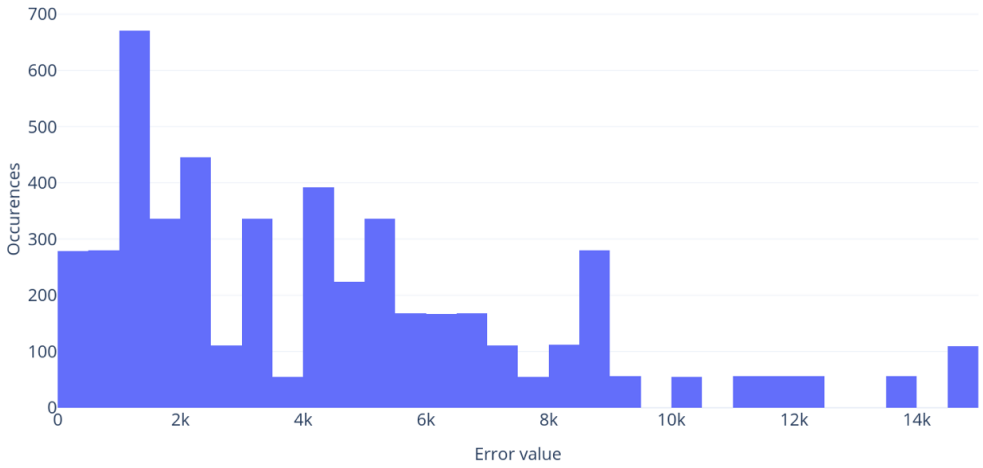

Figure 5. The distribution of prediction errors in case of group level measurements with the proposed method

With group level predictions the performance overhead is significantly lower, while advantages over the baseline are still maintained.

5.3. Performance Comparison

The main advantage of this approach can be easily seen with the following example case. Imagine that we want to use the clients from our measurement to run a PBC, with a maximized target hash-rate which is achieved in 80% (QoS metric) of the time. In other words, the question is what is the maximum capacity that we can say is achieved in 80% of the time.

Using the more realistic group-level approach, without prediction this value would be 681 452,2 hash/second, while with our method it would turn out to be 734 882,4 hash/second, which is a 9.3% improvement.

5.4. Conclusions

Based on the data observed, our solution to use a linear predictor based on the Robbins-Monroe algorithm proved to be advantageous in both client-level and group level predictions. Both the absolute value of prediction errors and the variance in the error distribution has decreased.
As the real-world applicability of client level prediction is limited, the main focus should be on group level predictions. In this case, the improvement (compared to the baseline) is even more considerable, while still having significantly lower performance overhead when compared to individual client level prediction approach.

6. Overview and Future Work

The research reported here deals with the open problems of private and permissioned blockchains (PBCs) and our goal was to find a solution to overcome some of these problems related to capacity planning and performance predictions. We formalized the problem and outlined the basic concepts of our solution by using a linear prediction of the available computational capacity. We investigated the performance of this solution in a simulated environment. The measurements are based on real measured data and we drew some conclusions regarding the observed outcome.

The measurements demonstrated that our proposed solution is more accurate in both approaches. In this way, better capacity planning and more efficient implementations for PBCs can be achieved. This may lead to better efficiency for PBCs, or larger systems with the same QoS, decreasing costs and environmental impact at the same time. We hope our work paves the way for the faster adaptation of the technology, ultimately leading to a better future, where trusted data is more accessible.

In the future, we intend to continue the refinement our method, one possible next step of our research is to improve the prediction accuracy even further, for example by using more advanced prediction methods, such as neural networks. Another welcome improvement would be on the side of validation, by using a real-world PBC instead of a simulated one, we could gain further insight to the workings of these systems, and hopefully come up with even better solutions.

Conflicts of Interest

The authors declare no conflict of interest.

ACKNOWLEDGMENTS

This work was performed in the frame of FIEK_16-1-2016-0007 project, implemented with the support provided from the National Research, Development, and Innovation Fund of Hungary, financed under the FIEK_16 funding scheme. It was also supported by the BME-Artificial Intelligence FIKP grant of EMMI (BME FIKP-MI/SC) and by the János Bolyai Research Fellowship of the Hungarian Academy of Sciences.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
[2] "Distributed ledger technology: Blackett review," GOV.UK. [Online]. Available: https://www.gov.uk/government/publications/distributed-ledger-technology-blackett-review. [Accessed: 20-Apr-2019].
[3] M. Jakobsson and A. Juels, "Proofs of work and bread pudding protocols," in Secure Information Networks, Springer, 1999, pp. 258–272.
[4] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the Security and Performance of Proof of Work Blockchains," in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS'16, Vienna, Austria, 2016, pp. 3–16.
[5] M. Vukolić, "The Quest for Scalable Blockchain Fabric: Proof-of-Work vs. BFT Replication," in Open Problems in Network Security, 2016, pp. 112–125.
[6] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," Ethereum Project Yellow Paper, vol. 151, pp. 1–32, 2014.
[7] E. Androulaki et al., "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," in Proceedings of the Thirteenth EuroSys Conference, New York, NY, USA, 2018, pp. 30:1–30:15.
[8] S. Pongnumkul, C. Siripanpornchana, and S. Thajchayapong, "Performance Analysis of Private Blockchain Platforms in Varying Workloads," in 2017 26th International Conference on Computer Communication and Networks (ICCCN), 2017, pp. 1–6.
[9] T. T. A. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, and K.-L. Tan, "BLOCKBENCH: A Framework for Analyzing Private Blockchains," in Proceedings of the 2017 ACM International Conference on Management of Data, New York, NY, USA, 2017, pp. 1085–1100.
[10] A. A. Pirzada and C. McDonald, "Establishing Trust in Pure Ad-hoc Networks," in Proceedings of the 27th Australasian Conference on Computer Science - Volume 26, Darlinghurst, Australia, Australia, 2004, pp. 47–54.
[11] P. Chatterjee, "Trust based clustering and secure routing scheme for mobile ad hoc networks," International Journal of Computer Networks and Communication, vol. 1, no. 2, pp. 84–97, 2009.
[12] A. Back, "Hashcash-a denial of service counter-measure," 2002.
[13] L. V. Jánoky, J. Levendovszky, and P. Ekler, "Application of Statistical Modeling and Participatory Computing for Private Blockchains," International Multidisciplinary Scientific GeoConference: SGEM: Surveying Geology & mining Ecology Management, vol. 18, pp. 149–156, 2018.
[14] L. V. Jánoky, J. Levendovszky, and P. Ekler, "New Solutions in the Design of Private and Permissioned Blockchains," in Proceedings of the Automation and Applied Computer Science Workshop 2019, pp. 154–163.
[15] H. Robbins and S. Monro, "A Stochastic Approximation Method," Ann. Math. Statist., vol. 22, no. 3, pp. 400–407, 1951.
[16] N. I. of S. and Technology, "Secure Hash Standard (SHS)," U.S. Department of Commerce, Feder Information Processing Standard (FIPS) 180-2 (Withdrawn), Aug. 2002.

AUTHORS

László Viktor Jánoky is a Ph.D. student at the Department of Automation and Applied Informatics of Budapest University of Technology and Economics, researching the new generation of distributed software systems.

Janos Levendovszky obtained his Ph.D. at the Budapest University of Technology and Economics and his DSc from the Hungarian Academy of Sciences. He is presently a full-time professor at the Budapest University of Technology and Economics and also vice-rector of Science and Innovation. His research area includes adaptive signal processing, networking, artificial intelligence, and algebraic coding theory.

Péter Ekler is an Associate Professor at the Department of Automation and Applied Informatics of Budapest University of Technology and Economics. His main research topic is mobile systems and networks.