A Simple Algorithm for Automatic Hopping among Pools in the Bitcoin Mining Network

Juan José García Chávez* & Carlo Kleber da Silva Rodrigues**

*Department of Electrical and Electronics, University of the Armed Forces - ESPE, Sangolqui, Ecuador, BRAZIL.
E-Mail: jjgarcia{at}espe{dot}edu{dot}ec
**Department of Electrical and Electronics, University of the Armed Forces - ESPE, Sangolqui, Ecuador, University Center of Brasília - UniCEUB, Brasília, DF, BRAZIL. E-Mail: carlokleber{at}gmail{dot}com

***Abstract*—**Herein it is proposed a simple algorithm for automatic hopping among mining pools in peer-to-peer networks using the Bitcoin protocol. The hopping ceases to occur when the best pool to be mined is found. The choice of the best pool is based on statistical data generated and collected during the online operation of the Bitcoin network. The deployment of this algorithm chiefly results in a bitcoin generation process significantly optimized, thereby making the whole system more effective. Experiments for validation and performance analysis of the proposed algorithm are based on mining networks built specifically for this purpose. Three performance metrics are assessed, namely duration of the block, payment by block in Satoshis, and yield. These metrics altogether are indeed an effective way to evaluate the performance of the bitcoin generation process. The major results show for instance that the bitcoin generation increases at 46.0 % comparing to when the mining is carried out in only one isolated pool. Lastly, general conclusions and possible avenues for future works are highlighted at the end of this article.

***Keywords*—**Bitcoin; Block; Blockchain; Hopping among Pools; Mining Pool.

***Abbreviations*—**Duration of the Block (BD); Payment by block in Satoshis (PS); Peer-to-Peer (P2P); Yield (Y).

I. INTRODUCTION

THE Bitcoin network is a peer-to-peer network (P2P) used to make financial transactions based on cryptocoins named as bitcoins [Nakamoto, 2; Barber et al., 5; Bamert et al., 7]. This network is managed by a protocol that receives the same name, Bitcoin protocol. The process of generating the bitcoins is called mining. A miner refers to the computer that is used for mining [Nakamoto, 2]. A mining pool is a group of miners who put together their computing capabilities to make a more efficient mining process.

Mathematically, the mining process involves solving the cryptographic problem named as the proof of work. It consists of finding the product of the Hash function, which is conformed by 256 bit and has a number of zeros in its beginning [Nakamoto, 2; Reid & Harrigan, 4; Donet et al., 15]. To find such a product, a set of transactions called a block is validated. Once the block is generated, it is added to the so called blockchain, which has all historical transactions and blocks that have been created since the launch of the Protocol in 2009 [Bohr & Bashir, 12; Back et al., 14].

If the mining is carried out in a unique pool, the bitcoin generation capacity is limited by the miners that conform the pool [Dorit & Adi, 11]. It takes a mining pool, with moments of bad luck, a longer time in the generation of a block than its average time. The efficiency of the mining is so deeply affected by the moments of bad luck. Furthermore, if the mining is carried out considering hops among pools, the mining capabilities belonging to other pools can be taken in conjunction as if it were only one, thereby potentially making the system operate more efficiently.

The chief motivation for this work is to improve the efficiency of the mining computers, reducing the drawbacks from the bad luck moments by means of hopping until finding a pool which may be more efficient. Within this context, this article has the objective to propose an algorithm which enables to determine the moments of bad luck as well as the right moment to hop among pools, taking into account the statistical data collected and generated during online operation. The validation of the algorithm is carried out by means of experiments grouped in three distinct mining scenarios. The first scenario comprises experiments without considering hopping among pools. The second scenario refers

to a situation where the automatic hopping is implemented taking into account a maximum of two pools. Lastly, the third scenario increases the number of pools which may be mined in order to evaluate the scalability of the solution.

It worth saying that three performance metrics are assessed in the above experiments, namely duration of the block, payment by block in Satoshis, and yield. These metrics altogether are indeed an effective way to evaluate the performance of the generation process. Among the major results, it may be noticed that the proposed algorithm makes the bitcoin generation increase at 46.0 % comparing to when the mining is carried out in only one isolated pool. The main contribution of this research is thus to provide a very simple solution that, if implemented, may significantly optimize the performance of the miners.

The remainder of this article is organized as follows. Section II presents the basis of the operation of the Bitcoin protocol and the mining process as well. Section III discusses some literature works concerning the subject herein explored. Section IV presents the algorithm for automatic hopping among pools. Section V shows the scenarios, results, and corresponding analysis. Finally, Section VI presents the final conclusions and suggestions for future works.

II. BITCOIN SYSTEM

Some specific terms that are of broad knowledge concerning the Bitcoin system are defined in what follows. The goal is to make the ideas to be discussed more easily understandable [Mier et al., 8; Kroll et al., 10].

- Address wallet: it is the address that enables to identify the user in the Bitcoin network. It allows the user to send and receive Bitcoins;
- Wallet program: it is the program that contains the public and private keys of a wallet, besides having the whole block chain in its original implementation;
- Destination/origin address: these are the addresses used to carry out concrete transactions;
- Public key: it is the wallet address;
- Commission: it is an incentive (in bitcoins) that can be given by the user to the miners so that greater priority can be used in his transactions.
- Private key: is the key known only by the user. It allows accessing to wallet to carry out transactions.

The algorithm 1 presents the step-by-step process of the practical use of the Bitcoin protocol. The point of view of a user will be used, where he accesses the network to perform a payment using bitcoins. It should be clear that these steps consider that the user already has: a wallet with a certain amount of bitcoins; the destination address, the origin address, and the amount of bitcoins to be sent.

Algorithm 1: Bitcoin use

Begin

Step 1: Login in the wallet using his private key.

Step 2: Add the destination address, which is the public key that identifies the destination wallet.

Step 3: Select the amount to be sent in bitcoins and, if desired, add a commission to the miners.

Step 4: Send the selected amount to the addressee.

Step 5: Wait for confirmation, carried out by the miners, to consider the transaction as valid. This may take a while, depending on the network status and the commission offered to the miners.

End.

III. RELATED WORK

Although there is a significant amount of works related to the Bitcoin network, there are not works specifically devoted to hopping among mining pools. Considering this fact, a discussion on some of the most important and recent references from the scientific literature follows. This is to give the reader a broad and general overview of the state of art of the Bitcoin network.

One of the first scientific work that deserves attention is the one of Back [1]. Therein are defined a number of terms used in the Bitcoin protocol. This work is mainly important due to the clear and thorough explanation of the so-called proof of work. This is the base of the mining process and, therefore, understanding its definition in details becomes mandatory for a proper exploration and improvement of the way the miners work. Additionally, the work of Back [1] is also a reliable reference for the topic transaction security, which is related to the mining process as well.

The original paper by Nakamoto [2] may be considered essential for any bitcoin related work. The author presents in detail the operation of the Bitcoin protocol. In other words, it comprises a very important reference because it enables a very elucidative overview of the mining process as a whole. Comparing to the work of Back [1], it may be noticed that the work of Nakamoto has a wider spectrum, since it involves all the concepts of cryptocoins and not only the concept of proof of work, as done by Back [1].

Eyal & Sirer [9] show that the Bitcoin protocol is not incentive-compatible, i.e., it does not incentivize miners to follow the protocol prescribed. They present an attack with which colluding miners obtain revenue larger than their fair share. From this, they then observe that the Bitcoin system ceases to be a decentralized currency. The implementation of our algorithm avoids the problem pointed out by Eyal & Sirer [9], since several pools are going to be mined and not just only one anymore.

Finally, another interesting article which is indirectly related to Bitcoin mining is the work of Johnson et al., [13].

The authors analyze hypothetical attacks of service degeneration to pools (DDoS) and how to produce an attack with these characteristics would benefit those attacking the pools, disregarding the big pools. Similarly to above, the implementation of our algorithm minimizes the effects of such attacks, since there is hopping among distinct pools.

IV. ALGORITHM FOR AUTOMATIC HOPPING

In this section, Algorithm 2 is presented. It refers to the proposal for the automatic hopping among mining pools. It is worth mentioning that this description takes into account the statistical data generated for hopping among pools.

Algorithm 2: Automatic hopping among mining pools

Begin

Step 1: Capture the time it takes the pools to generate the last blocks.

Step 2: Calculate the average and the standard deviation of the generation time of every pool considering the data obtained in the Step 1.

Step 3: The pool with an actual time closer to the average and that also presents the smaller standard deviation, regarding its average, is to be the one chosen.

Step 4: Monitor all the pools that have been considered.

Step 5: When the generation time of the pool exceeds the average plus the standard deviation, the average times and standard deviations of the other pools are calculated.

Step 6: The pool with an actual time closer to the average and that also presents the smaller standard deviation, regarding its average, is the one to be chosen.

Step 7: Hop to the pool selected in Step 6.

Step 8: Return to Step 4.

End.

Note that the algorithm shown above takes into account the statistical data collected and generated during online operation over the pools being mined. More specifically, the statistical data allow computing the average generation time and standard deviation, thereby enabling to determine which pool may be more efficiently mined during online operation. It is worth mentioning that, to the best of knowledge, there is no similar algorithm considering the approach herein adopted to improve the mining process.

The process above may be understood as a kind-of pool supervision that enables to determine when the pool is in moments of bad luck. This is when a hop has to take place, making it possible to take advantages of a more efficient pool. In Section V, the performance of the algorithm is evaluated, comparing cases where there is the employment of the algorithm to those where there is not.

V. PERFORMANCE EVALUATION

5.1. Scenarios and Metrics

In this subsection, three scenarios are shown: Scenarios 1, 2 and 3. They are used in the experiments for the performance evaluation of the proposed algorithm. The total capacity of the computer equipment in all scenarios is of 460GHs (Giga hash per second) in mining, with specific chips ASIC (Application Specific Integrated Circuit). Besides, there is a computer used as server, where the proposed algorithm is executed.

Figure 1 depicts Scenario 1, which is constituted by two pools. The user initially accesses pool 1 and, after a determined time, accesses pool 2. This time is set as a day, since it is a reasonable time to obtain trustable results. It should be clear that pool 1 is referred to the Slush Pool [Smith, 6], and pool 2 is referred to GHASH.IO pool [Palatinus, 3]. The Slush Pool was the first one to operate since the launch of the Bitcoin system, and the GHASH.IO was the pool with the biggest capacity in the Bitcoin network when the experiments of this work were carried out. In this scenario (Scenario 1) there are not new implementations. Only the data being generated by the operation of two different and independent pools are collected.

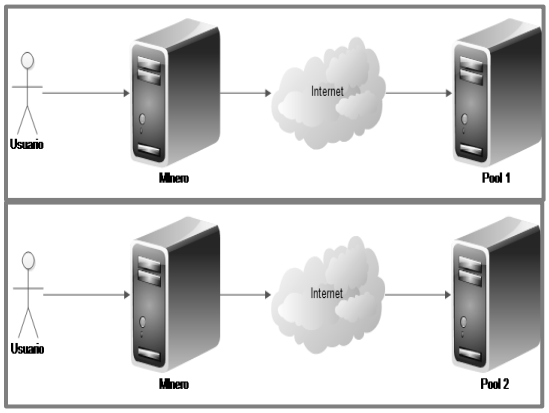


Figure 1: Scenario 1 (Comparison between Two Pools at the same time)

Figure 2 presents the second scenario, which is constituted by the two pools previously used. The difference is that the implementation for the automatic hopping exists. Data of the operation of the two different pools is collected. Finally, Figure 3 shows the third scenario, which also possesses the implementation of the automatic hopping among n mining pools. In this case, data coming out of the operation of $n = 6$ different pools is collected. This value of n suffices for the goal of this work. Finally, the performance metrics used to analyze the new protocol are detailed in Table 1.

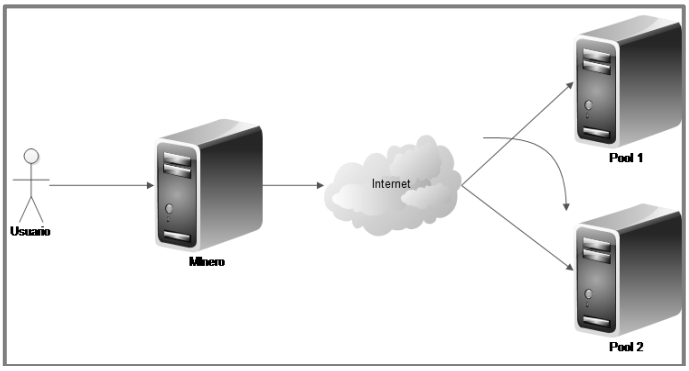


Figure 2: Scenario 2 (Implementation of the Algorithm for the Automatic Hopping for Two Pools)

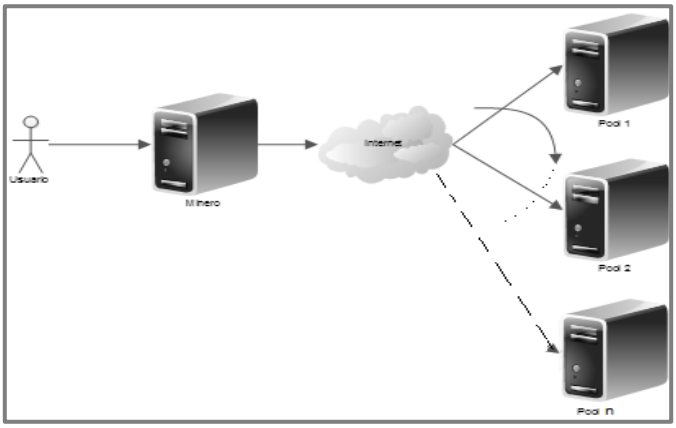


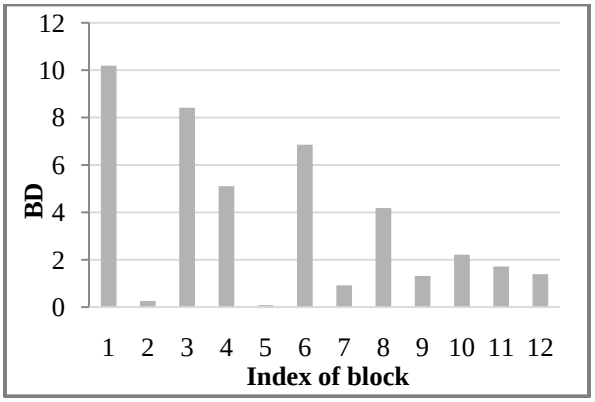
Figure 3: Scenario 3 (Implementation of the Automatic Hopping among Pools for n Pools)

Table 1: Metrics

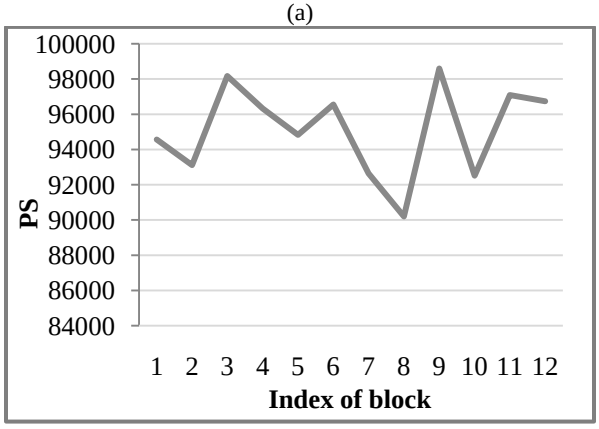
Metric	Definition
Duration of the block (BD)	It is the time that the pool takes to find a block.
Payment by block in Satoshis (PS)	It is the payment that is given to the pool when it finds a block. Satoshis is the minimum amount of a Bitcoin, i.e., 1 Satoshi = 0.00000001
Yield (Y)	It is the quantity in Satoshis by hour produced by the miners in different scenarios. In all cases the metric Y is computed by: $Y = \frac{sum\ of\ all\ PS}{Sum\ of\ all\ BD}$

5.2. Results and Analysis

Figures 4, 5 and 6 present the results obtained for the metrics in Scenario 1. By the results, it can be concluded what follows. For the metric BD (see Figures 4(a) and 5(a)), it may be see that pool 2 is more efficient than pool 1; however, for the metric PS (see Figures 4(b) and 5(b)), the situation changes, since pool 1 becomes more efficient due to the higher payments. So, to solve this contradiction, it comes as a need to compute the metric Y to determine the more advantageous pool. The result (see Figure 6, Scenario 1) shows that pool 1 is more efficient.

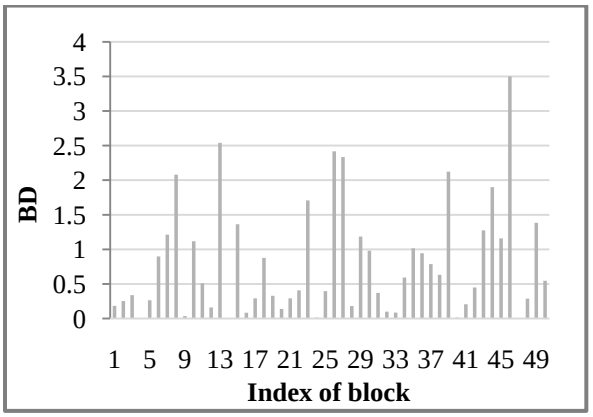


(a)

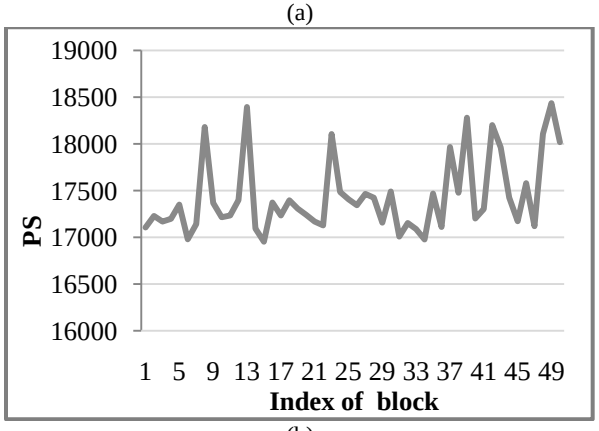


(b)

Figure 4: Scenario 1 in Slush Pool: (a) Duration of the Block in Hours (BD); (b) Payment by Block in Satoshis (PS)



(a)



(b)

Figure 5: Scenario 1 in GHash.io.: (a) Duration of the Block in Hours (BD); (b) Payment by Block in Satoshis (PS)

Now, from the results shown in Figure 10, it can be seen that an increase at the value of Y in Scenario 2 is obtained. More precisely, 7% with respect to the Slush Pool, while 30% with respect to GHASH.IO. When the number of pools is incremented to 6 in Scenario 3, the value of the metric Y increases: at 12% with respect to the Scenario 2; at 19% with respect to the Slush Pool; and at 46 % respect to the GHASH.IO.

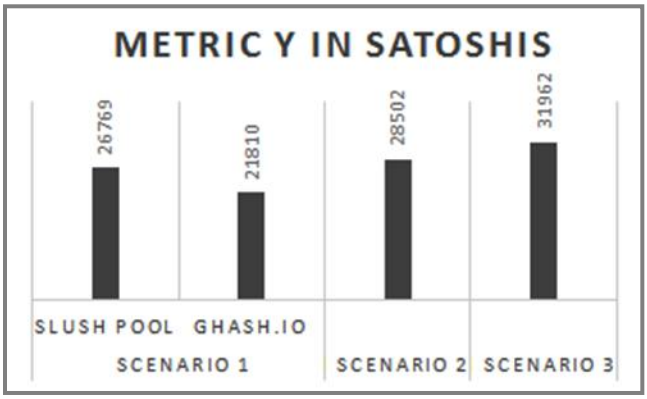
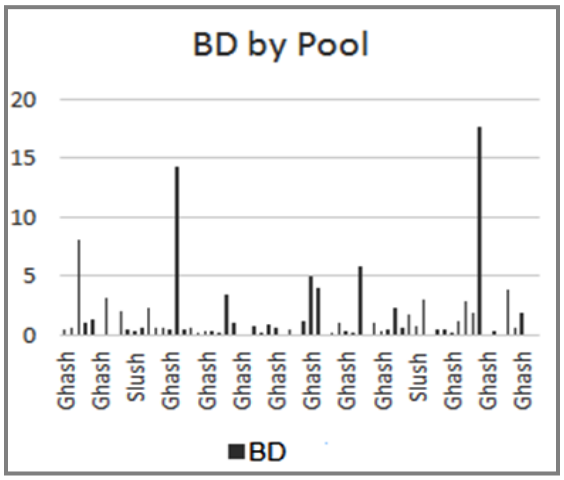
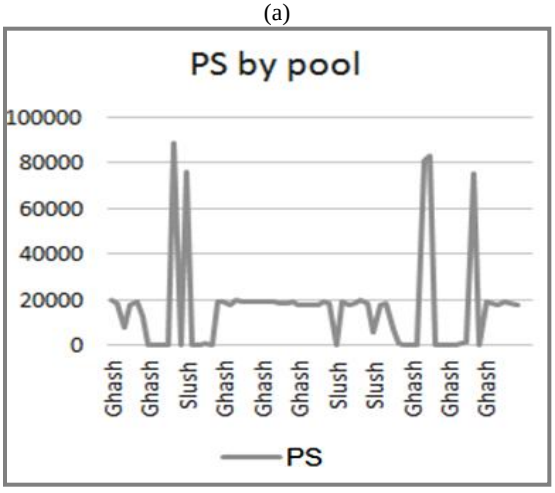


Figure 6: Yields in the Different Scenarios



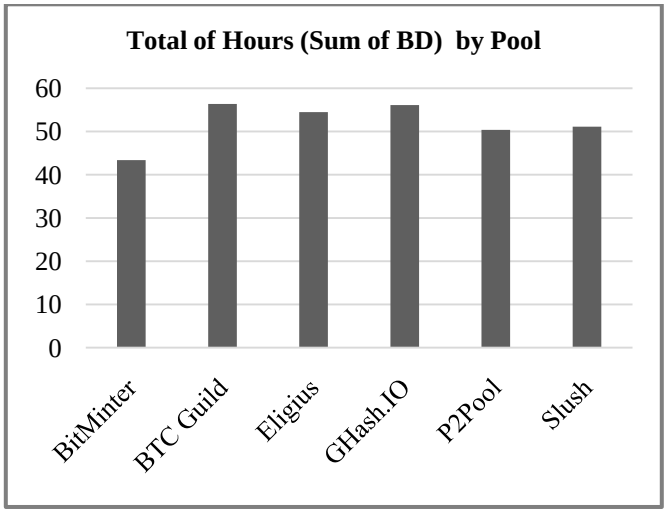
(a)



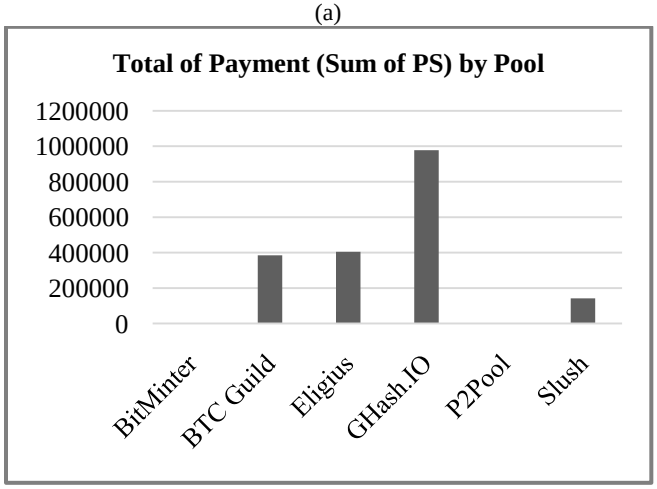
(b)

Figure 7: Scenario 2: (a) Duration of the Block in Hours (BD); (b) Payment by Block in Hours (PS)

Figures 7 and 8 present the results obtained for the metrics in Scenarios 2 and 3, respectively. In the former scenario, the generation time between both pools selected can be independently measured; they depend on the chosen pool and the way each pool pays. In the latter scenario, it can be concluded that a certain preference exists to the pools that have a greater speed, excluding the smaller ones that can give a better value of Y, as observed in Scenarios 1 and 2, respectively (see Figure 6). This could be a motivation for subsequent works.



(a)



(b)

Figure 8: Scenario 3: (a) Hours of the Blocks Mined by Pool; (b) Payment by Pool

From the above experiments and analysis, the more positive overall observation is that hopping among distinct mining pools is a truly advantageous solution, since it enables a more efficient bitcoin generation process. Additionally, the proposed solution does scales well, i.e., the more distinct mining pools, the more efficient the overall system becomes. The algorithm complexity is also of very feasible implementation since the statistical data to be considered for selecting the next pool are generated in a very spontaneous way during online operation. On the other hand, the pools with lower capacity, within the candidates that may be selected, are usually neglected, i.e., are never selected to be the next one. This drawback is intentionally left as future work.

VI. Conclusions and Future Works

This article proposed a simple algorithm for automatic hopping among mining pools in peer-to-peer networks using the Bitcoin protocol. The hopping ceased to occur when the best pool to be mined was found. The choice of the best pool was based on statistical data generated and collected during the online operation. Experiments for validation and performance analysis were based on mining networks built specifically for this purpose. Three performance metrics were assessed, namely duration of the block, payment by block in Satoshis, and yield.

Within the most important conclusions, the following may be for instance highlighted: (1) the proposed algorithm was really efficient and of feasible implementation; (2) the results indicated that the generation of bitcoins increases at 46.0 % comparing to when mining in a same single pool; and (3) when many pools are employed, the algorithm mostly works among only those with a higher performance only (i.e., those showing higher speed to find a block).

Considering possible future works, the following one may be suggested. First, the proposed algorithm may be employed as a solid baseline to develop other proposals aiming at even more efficient hops among pools, considering other different technical aspects. For instance, P2P networks demand more participants so as to have more reliability and stability. Then modifying the proposed algorithm to include pools of lower capacity could still optimize the overall Bitcoin system performance. Lastly, the proposed algorithm may be used to design and implement an intermediate pool which allows the automatic hopping among other pools, so that some additional performance could be potentially achieved.

References

[1] A. Back (2002), "Hashcash - A Denial of Service Counter-Measure", URL: http://www.hashcash.org/.

[2] S. Nakamoto (2008), "Bitcoin: A Peer–to–peer Electronic Cash System", URL: http://www.bitcoin.org/bitcoin.pdf.

[3] M. Palatinus (2010), "Slush Pool", URL: http://mining.bitcoin.cz/.

[4] F. Reid & M. Harrigan (2011), "An Analysis of Anonymity in the Bitcoin System", *Proc. of the 1st Workshop on Security and Privacy in Social Networks (SPSN'11) at SocialCom 11 IEEE*, Pp. 1318–1326.

[5] S. Barber, X. Boyen, E. Shi & E. Uzun (2012), "Bitter to Better-How to Make Bitcoin a Better Currency", *Financial Cryptography and Data Security - 16th International Conference*, Pp. 397–414.

[6] J. Smith (2013), "GHASH.IO", URL: https://ghash.io/.

[7] T. Bamert, C. Decker; L. Elsen, R. Wattenhofer & S. Welten (2013), "Have a Snack, Pay with Bitcoins", *IEEE Thirteenth International Conference on Peer-to-Peer Computing (P2P).*

[8] I. Mier, C. Garman, M. Green & A.D. Rubin (2013), "Zerocoin: Anonymous Distributed E-Cash from Bitcoin", *2013 IEEE Symposium on Security and Privacy (SP)*, Pp. 397–411.

[9] I. Eyal & E.G. Sirer (2013), "Majority is not Enough: Bitcoin Mining is Vulnerable", *Financial Cryptography and Data Security*, Vol. 7859, Pp. 436–454.

[10] J.A. Kroll, I.C. Davey & W. Edward (2013), "The Economics of Bitcoin Mining or Bitcoin in the Presence of Adversaries", *Workshop on the Economics of Information Security.*

[11] R. Dorit & S. Adi (2013), "Quantitative Analysis of the Full Bitcoin Transaction Graph", *Financial Cryptography and Data Security*, Vol. 7859, Pp. 6–24.

[12] J. Bohr & M. Bashir (2014), "Who Uses Bitcoin? An Exploration of the Bitcoin Community", *Privacy, Security and Trust (PST)*, Pp. 94–101.

[13] B. Johnson, A. Laszka, J. Grossklags, M. Vasek & T. Moore (2014), "Game - Theoretic Analysis of DDoS Attacks Against Bitcoin Mining Pools", *Financial Cryptography and Data Security*, Vol. 8437, Pp. 72–83.

[14] A. Back, M. Corallo, L. Dashjr, M. Friedenbach, G. Maxwell, A. Miller, A. Poelstra, J. Timón & P. Wuille (2014), "Enabling Blockchain Innovations with Pegged Sidechains", URL: http://www.opensciencereview.com/papers/123/enabling-blockchain-innovations-with-pegged-sidechains.

[15] J.A.D. Donet, C. Perez-Sola & J. Herrera-Joancomart (2014), "The Bitcoin P2P Network", *Financial Cryptography and Data Security*, Vol. 8437, Pp. 87–102.

Juan José García Chávez. He was born on December 29, 1986. Studied bachelor in Industrial Electricity at the Salesian Technical School "Don Bosco" and is currently studying Electronic Engineering in Networks and Data Communication at the University of the Armed Forces – ESPE. Also, he studies at CISCO Networking Academy. His areas of research interest include computer networks, open and free software/hardware development and Bitcoin business and development. Now, He gives technical support in Bitcoin Ecuador Community.

Carlo Kleber da S. Rodrigues received the B.Sc. degree in Electrical Engineering from the Federal University of Paraiba in 1993, the M.Sc. degree in Systems and Computation from the Military Institute of Engineering (IME) in 2000, and the D.Sc. degree in System Engineering and Computation from the Federal University of Rio de Janeiro (UFRJ) in 2006. Currently he is Military Assessor of the Brazilian Army in Ecuador, Professor at the Armed Forces University (ESPE) in Ecuador, and Professor at the University Center (UniCEUB) in Brazil. His research interests include the areas of computer networks, performance evaluation, and multimedia streaming.