I.J. Wireless and Microwave Technologies, **2018, 5, 1-9**

Available online at http://www.mecs-press.net/ijwmt

A Proof of Work: Securing Majority-Attack In Blockchain Using Machine Learning And Algorithmic Game Theory

Somdip Dey

School of Computer Science and Electronic Engineering, University of Essex, U.K.

Email: somdip.dey @ essex.ac.uk

Abstract

Blockchain's vast applications in different industries have drawn several researchers to pursue extensive research in securing blockchain technologies. In recent times we could see several institutions coming together to create consortium based blockchain networks such as Hyperledger. Although for applications of blockchain such as Bitcoin, Litcoin, etc. the majority-attack might not be a great threat but for consortium based blockchain networks where we could see several institutions such as public, private, government, etc. are collaborating, the majority-attack might just prove to be a prevalent threat if collusion among these institutions takes place. This paper proposes a methodology where we can use intelligent software agents to monitor the activity of stakeholders in the blockchain networks to detect anomaly such as collusion, using supervised machine learning algorithm and algorithmic game theory and stop the majority attack from taking place.

Index Terms: Computer Security, network, blockchain, machine learning, algorithmic game theory, majority attack, anomaly detection.

I. Introduction

When Satoshi Nakamoto [2] released the technology named Bitcoin, he revolutionized the industry not because he has invented a new currency system, which do not require intervention of institutional mediator while transferring money from one entity to another, but because he has gifted one of the most disruptive technology, which has come to life in decades. With the introduction of Bitcoin, Blockchain got introduced to the world, which is a digital ledger in which all transactions are recorded chronologically and publicly. But the application of blockchain is not just limited to cryptocurrencies [3, 4] such as Bitcoin and have proved to be useful in tracking ownership, provenance of documents, digital assets, physical assets, voting rights, etc.

According to [5, 39, 40, 41, 42] blockchain network is fundamentally of three types as follows:

Public: Everyone in the network can check and verify the transactions made. The network is also open to

anyone who want to participate in the consensus process.

Private: This network has strict restrictions on data access and nodes (user/entity) have restricted access to specific block chains, which is monitored by a governing body.

Consortium: Nodes in the network can form partnership with businesses or other authorities and the network may be public or private. So this could be seen as a hybrid approach as partly decentralized.

As we can see the rise in use of blockchain technologies, we can also see rise of security issues such as 'Double-Spending' especially in the Majority Attack [2, 5, 6, 7, 8]. The majority attack is carried out by a group of individuals/entities in the decentralized environment, who colludes to take control over the ledger to gain profit from it. In this proposed methodology, a novel approach of using Algorithmic Game Theory concepts and Machine Learning techniques is used to reduce the chances of collusion in the decentralized system to gain advantage over other miners so that the system can be as fair as possible. In section 2, some background theory regarding blockchain and economy of double-spending is discussed. In section 3, the proposed methodology is discussed. Finally the papers ends with some discussion in section 4 and conclusion.

2. Background Theory and Related Work

In this section we will visit the concepts of doublespending and the majority attack in Blockchain. Later in this section we will also discuss the economy of the attack being performed and how Game Theory is applicable to security implications in blockchain as well.

Blockchain technology are so popular at the moment because of its design features, which are composed of six key elements as follows [2, 5, 37, 38]:

Decentralized: Blockchain data could be recorded, stored and updated distributedly without depending on a central authority or node.

Transparent: Data recorded and stored are transparent and are visible thus leveraging trust among its users.

Open Source: The source code as well as the most of the blockchain dependent systems are open to view, free to use and provide the ease of extension for other applications.

Autonomous: Blockchain updates are consensus based and thus data could be updated securely from a single user to the whole system. This feature provides autonomy to the system to update data securely.

Immutability: All data in the blockchain are reserved forever and can't be modified unless a single entity/user or a group of users collude and take over more than 51% of the computing resources of the system (This is called *majority attack* [6, 7]).

Anonymity: Blockchain also provides anonymity to its users and make the system more trust worthy by only using the users' blockchain addresses instead of their personal information.

Although blockchain's design make it very suitable for several applications while providing trust to its users but as with any networked system on this planet, blockchain is no exception to security attacks [9 - 33] and hacks. One such noteworthy security issue is the majority attack where an entity or user could take control of

the node and use it for self benefit if the attack is performed properly. In the next subsection we will discuss elaborately about this security issue.

2.1. Blockchain, Double-Spending & Majority Attack

A double-spending attack [2, 5, 6, 7] in blockchain means the attacker has to convince the merchant that a transaction has been confirmed and then convince the entire network to approve some other transaction, which will lead to the attacker keeping both the money and the service (goods) from the merchant whereas the merchant would be left with neither the money or the service. This problem in synchronization is solved by proof-of-work, which is a computational effort consisting of hashes to acknowledge the groups of transactions, also known as blocks. For a transaction to be valid, sufficient work has been done to acknowledge that the block contains it. Since, validation of blocks require computational effort to do so, this also gives rise to another issue, what if the attacker has substantial computational power at its disposal? All the attacker has to do is mine a blockchain privately till the length of the chain becomes longer than the chain mined by the honest network, and release this private blockchain for confirmation when it is appropriate. In Rosenfeld's paper [6] the probability of the attacker succeeding in his attack is discussed. If we consider z as the number of blocks by which the honest network has advantage over the attacker then $z = n - m$, where n is the number of blocks in the chain on top of the one where fork started for the honest network, whereas m is the number of blocks in the chain on top of the fork which the attacker has built. Before we discuss the probability of having advantage over the attacker, let us consider the following assumptions:

The total hashrate of the attacker and honest network is constant. They have a hasrate of H combined, of which pH belongs to honest network and qH belongs to the attacker, where $p + q = 1$. The mining difficulty is constant, such that the time taken to find a block with H hashrate is T_o. There are two possibilities of *double-spending attack*, which is either the attack succeeds or it fails, as follows:

$$z_{i+1} = \begin{cases} z_i + 1 & with\ probability\ p \\ z_i - 1 & with\ probability\ q \end{cases}$$

If we consider a_z to be the probability of the attacker succeeding in the attack then we can arrive at the following equation:

$$a_z = pa_{z+1} + qa_{z-1} \quad \text{…. (1)}$$

And if we solve this using the boundary condition and the notion $p + q = 1$ then we can conclude:

$$a_z = \min\left(q/p, 1\right)^{\max(z+1,0)}$$

$$= \begin{cases} 1 & if\ z < 0 \,||\, q > p \\ (q/p)^{z+1} & if\ z \geq 0 \,||\, q \leq p \end{cases} \quad \text{….(2)}$$

If we assume n number of blocks are found by the honest network and $m + 1$ number of blocks are found by

the attacker during this time period then the probability (r) of double-spending to succeed when the merchant waits for n confirmations using the equation (2) is:

$$r = \sum_{m=0}^{\infty} P(m) a_{n-m-1}$$

$$= \begin{cases} 1 - \sum_{m=0}^{n} \binom{m+n-1}{m} (p^n q^m - p^m q^n) & if\ q < p \\ 1 & if\ q < p \end{cases}$$

….(3)

In the study [6], it is proved that as the number of confirmations by the honest network increased, the success rate of the attack decreased but no matter how many confirmations by the honest network has succeeded, the attack will always succeed if the hashrate of the attacker approached 50% of the total network hashrate, which means q greater or equal 0.5.

This proves that an attacker with more computing power at its disposal might prove to be a key factor in succeeding in the attack. This particularly raises security concerns in Consortium Blockchain [5, 7, 8] such as Hyperledger, where we can see involvement of several companies or business entities. Whoever in the Hyperledger network holds the maximum computing power, can always get a competitive advantage over its competitors while performing business transaction over the network.

With Proof of Work, more CPU/GPU power is required in checking hashes of each block in the blockchain. Because of this mechanism, more and more business entities would like to join in this mining process, which would create "*mining pools*", and once the mining pool holds 51% computing power, then it would take control of the blockchain. Therefore, by taking control what it can do is [5, 8]:

1. Modify the transaction data, which can lead to double spending attack
2. To stop the block verification transaction
3. To stop miners mining any available block

Now, in order to make Hyperledger fair for every business entity/institution involved in the network, it is highly desirable that the Majority Attack do not succeed or not take place at all.

2.2. Economy of Double-Spending

In the study by Rosenfeld [6], it was found that the number of confirmations required to keep the success rate of the attacker (double-spending) below 10%, 1% and 0.1%, are 2, 4 and 6 respectively. In addition, we have already seen that once the attacker's hashrate reaches 50% of the total network hashrate then the number of confirmations required reaches infinity, which means no amount of confirmation can defeat the attack. Taking this into account, we also have to consider the likelihood of the attack being performed in reality. If value of the commodity being exchanged is assumed to have a value of v and the attacker has mined o number of blocks where each block has a value of B, then if the attack succeeds the attacker will gain v, where if the attack fails then the attacker will lose $v + oB$. Therefore, if we consider the two possibilities, the payoff (s) for the attacker is as follows:

$$s = \begin{cases} v & if\ q \geq 0.5 \\ -(v + oB) & if\ q < 0.5 \end{cases}$$

where, q is the hashrate of the attacker ….(4)

And in order to carry on with the attack the value of v has to be significant. This payoff (s) will prove to be useful in portraying the security implication in the light of Game Theory, and how decisions can be made to classify whether an attack is taking place or not.

3. Proposed Methodology

In section 2.B we have already seen that payoff (*s*) for the attacker can only have two possibilities: succeed or fail. This is where Game Theory [1] comes into account. But before we get into the concept let us define few terminologies of Game Theory in this context as follows:

• *Self-Interested Agents*: This can be any entity such as a person, business or any other institution in the blockchain network with their own preferences and utility. This also includes honest entities and attacker(s).

• *Player*: Each Self-Interested Agent who are participating in the blockchain network. Let us assume that there are N players where N = (1,….., n) is a finite set of n, indexed by i.

• *Action*: Action taken by each player based on their preferences and utility. And let us assume that set of actions taken by the player i is Ai where Ai = (a_1,…., a_n).

• *Payoff*: The reward, which each player receives.

Now, if we consider the equation (4) then we can see the attacker would want to maximize the probability of getting a payoff of v instead of loosing $v + oB$. Therefore, we can extend the same equation (4) to derive the utility function/ payoff function as follows:

$$u(a) = \begin{cases} v & if\ q \geq 0.5 \\ -(v + oB) & if\ q < 0.5 \end{cases}$$

….(5)

where, *u* is utility, *a* is the action taken by the attacker, *q* is the hashrate of the attacker, *v* is value of commodity/service by the merchant, *o* is number of blocks mined, *B* is value of each block.

This utility function (Eq. 5) will govern the decision on whether an attack is bound to happen or not by the attacker based on the value of the commodity/service. And in order to keep the blockchain network safe from the Majority Attack we should focus on this function.

We can feed this utility function to Supervised Machine Learning algorithms to classify whether an attack is

likely to take place or not. If the attack is likely to take place then set of rules should be implemented by the system to either prevent the blockchain confirmation from the attacker(s) or to prevent confirmation of the whole transaction till a new fair transaction is performed again i.e. no payoffs for anyone, in order to ensure fairness and legitimate transactions being confirmed in the network.

In order to achieve this, an intelligent agent is implemented in the application layer of the blockchain network system, which would have two distinct parts:

1) Based on the past transactions of the stakeholders the probability of each stakeholder to defect

2) Based on the current value of the commodity/service being sold in the current transaction the probability of the stakeholder(s) to attack through majority attack

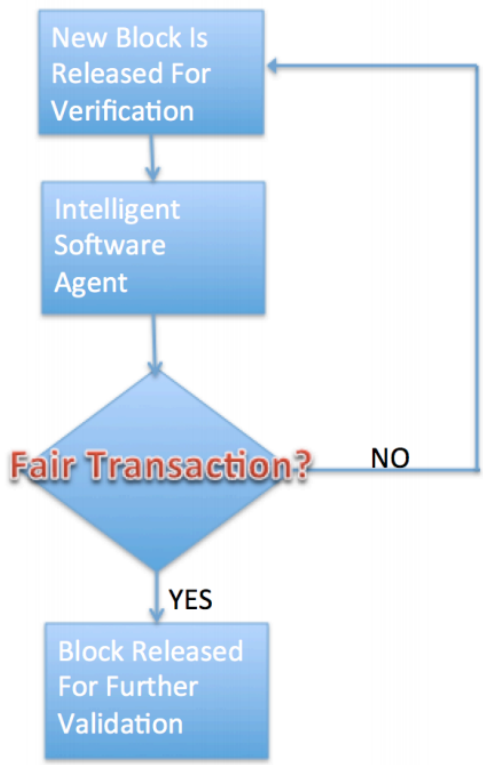

Fig. 1: Proof-of-Work (The Proposed Methodology)

In Fig. 1, we can see the workflow of the proposed methodology where after the new block is released in the network for the consensus purpose by the stakeholders (including attacker and other players), the intelligent software agent in the application layer of the network uses the utility function (Eq. 5) to classify the motive of the stakeholders and the value of the current service being sold in the transaction. If the motive of the stakeholders is deemed to be malicious in nature with the intent of collusion to perform a majority attack then the transaction is cancelled and all the stakeholders are requested for a new transaction instead.

4. Discussion

In the proposed methodology we have discussed about the utility function (Eq. 5) based on the value of the service or commodity being sold in the current transaction. Here, the commodity does not have to be something that has a tangible value in the network, rather it can have some personal attachment or importance to the stakeholder(s). In that case the intelligent agent needs to deduce the level of attachment or importance of the commodity or service being handled in the transaction in order to calculate the utility function and then the probability of the majority attack from taking place. To make the proposed methodology effective, it should be implemented in the application layer of the network where all the events on each node are recorded by the intelligent agent and used later to make decision.

Conclusion

As blockchain technology becomes more and more popular, we can see emergence of several variations of such consensus based distributed ledger systems where majority-attack can become more proficient. In order to prevent such malicious activity in the consensus based distributed ledger systems we can utilize some variations of the Proof-of-Work proposed in this paper. Although this is a work in progress and in its preliminary stage, the proposed Proof-of-Work will be extended to provide more holistic approach to such issues faced in the system.

Acknowledgment

The author wish to thank his colleagues at ReMe Basket Ltd. and Codeepy UK Pvt. Ltd. for their support. He would also like to thank his parents, Soma Dey and Sudip Dey, for their continued support and faith on the author's capabilities. This work was supported in part by a grant from Codeepy Pvt. Ltd. with reference CDPY/2018/1.

References

[1] Nisan et al. (2007), Algorithmic Game Theory. Cambridge University Press.
[2] S. Nakamoto, 'Bitcoin: A Peer-to-Peer Electronic Cash System'. Available at https://bitcoin.org/bitcoin.pdf (Accessed: 30th January, 2018).
[3] M. Staples, (2016) 'Blockchain is useful for a lot more than Bitcoin', The Conversation. Available at http://theconversation.com/blockchain-is-useful-fora-lot-more-than-just-bitcoin-58921 (Accessed: 30th January, 2018).
[4] (2018) 'How could blockchain be used in the enterprise', Computer World UK. Available at https://www.computerworlduk.com/galleries/securit y/how-could-blockchain-be-used-the-enterprise3628558/ (Accessed: 30th January, 2018).
[5] I. Lin, T. Liao, (2017) 'A Survey of Blockchain Security Issues and Challenges', International Journal of Network Security, Vol. 19, No. 5, pp. 653-659.
[6] M. Rosenfeld, (2014) 'Analysis of hashrate-based double-spending', ArXiv CoRR. Available at https://arxiv.org/pdf/1402.2009.pdf (Accessed: 30th January, 2018).
[7] N. T. Courtois, (2014) 'On The Longest Chain Rule and Programmed Self-Destruction of Crypto Currencies', ArXiv CoRR. Available at https://arxiv.org/pdf/1405.0534.pdf (Accessed: 30th January, 2018).
[8] I. Eyal, E. G. Sirer, (2013) 'Majority is not Enough: Bitcoin Mining is Vulnerable', ArXiv CoRR. Available at https://arxiv.org/pdf/1311.0243.pdf (Accessed: 30th January, 2018).
[9] T. Gopal, M. Subbaraju, R. v. Joshi, S. Dey, "MAR(S)2: Methodology to Articulate the Requirements for Security In SCADA", Proceedings of IEEE 2014 - Fourth International Conference on Innovative Computing Communication Technology (INTECH 2014), pp. 103-108.
[10] S. Dey, S. S. Ayyar, SB Subin, P.K. A. Asis, "SD-IES: An Advanced Image Encryption Standard", IEEE 2013 7th International Conference on Intelligent Systems and Control, Coimbatore, India, January 2013

[11] S. Dey, "New Generation of Digital Academic-Transcripts using encrypted QR Code™", IEEE 2013 International Multi Conference on Automation, Computing, Control, Communication and Compressed Sensing (iMac4s 2013), Kerala, India, March 2013
[12] S. Dey, S. Agarwal, A. Nath, "Confidential Encrypted Data Hiding and Retrieval Using QR Authentication System", IEEE International Conference On Communication System and Network Technologies 2013 (CSNT-2013), Gwalior, India, April 2013
[13] S. Dey, "SD-EQR: A New Technique to Use QR CodesTM in Cryptography", International Conference on Emerging Trends in Computer and Information Technology (ICETCIT 2012), Coimbatore, India, May 2012
[14] S. Dey, "SD-REE: A Cryptographic Method To Exclude Repetition From a Message", The International Conference on Informatics & Applications (ICIA 2012), Kuala Terengganu, Malaysia, June 2012
[15] S. Dey, "SD-EI: A Cryptographic Technique To Encrypt Image", IEEE 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec2012), Kuala Lumpur, Malaysia, June 2012
[16] S. Dey, "SD-AEI: An Advanced Encryption Technique For Images", IEEE 2012 Second International Conference on Digital Information Processing and Communications (ICDIPC2012), Klaipeda City, Lithuania, July 2012
[17] S. Dey, J. Nath, A. Nath, "Modified Caesar Cipher method applied on Generalized Modified Vernam Cipher method with feedback, MSA method and NJJSA method: STJA Algorithm", 8th International Conference on Foundations of Computer Science (FCS12), Las Vegas, USA, July 2012
[18] S. Dey, Joyshree Nath, Asoke Nath, "A New Technique to Hide Encrypted Data in QR CodesTM", 3th International Conference on Internet Computing (ICOMP12), Las Vegas, USA, July 2012
[19] S. Dey, "SD-MARC: A New Multi-Processor Architecture", 12th International Conference on Computer Design (CDES12), Las Vegas, USA, July 2012
[20] S. Dey, "A New Technique to Use a Parallel Compiler for Multi-core Microcontrollers", 12th International Conference on Computer Design (CDES12), Las Vegas, USA, July 2012
[21] S. Dey, "SD-AREE: A New Simple Cryptographic Method to Exclude Repetition from a Message to be Encrypted", IEEE 2012 Third International Conference on Computing, Communication and Networking technologies (ICCCNT12), Karur, India, July 2012
[22] S. Dey, A. Nath, "Modern Encryption Standard (MES) version-I: An Advanced Cryptographic Method", IEEE 2012 2nd World Congress on Information and Communication Technologies (WICT-2012), Trivandrum, India, October 2012
[23] S. Dey, "SD-C1BBR: SD-Count-1-Byte-Bit Randomization: A New Advanced Cryptographic Randomization Technique", IEEE 2012 2nd World Congress on Information and Communication Technologies (WICT- 2012), Trivandrum, India, October 2012
[24] S. Dey, M. S Nair. "Design and Implementation of a Simple Cache Simulator in Java to Investigate MESI and MOESI Coherency Protocols". International Journal of Computer Applications 87(11): 6-13, February 2014. Published by Foundation of Computer Science, New York, USA.
[25] S. Dey, "SD-AREE: An Advanced Modified Caesar Cipher Method to Exclude Repetition from a Message", International Journal of Information and Network Security (IJINS), Volume 1, Issue 2, pp. 67-76, 2012.
[26] S. Dey, K. Mondal, J. Nath, A. Nath, "Advanced Steganography Algorithm Using Randomized Intermediate QR Host Embedded With Any Encrypted Secret Message: ASA_QR Algorithm", IJMECS (International Journal on Modern Education and Computer Science), 2012.
[27] S. Dey, J. Nath, A. Nath, "An Advanced Combined Symmetric Key Cryptographic Method using Bit Manipulation, Bit Reversal, Modified Caesar Cipher (SD-REE), DJSA method, TTJSA method: SJA-I Algorithm", International Journal of Computer Applications 46 (20): 46-53, May 2012. Published by Foundation of Computer Science, New York, USA.
[28] S. Dey, J. Nath, A. Nath, "An Integrated Symmetric Key Cryptographic Method Amalgamation of TTJSA Algorithm, Advanced Caesar Cipher Algorithm, Bit Rotation and Reversal Method: SJA Algorithm", IJMECS, vol.4, no.5, pp.1-9, 2012.
[29] S. Dey, "SD-AREE-I Cipher: Amalgamation of Bit Manipulation, Modified VERNAM CIPHER & Modified Caesar Cipher (SD-AREE)", IJMECS, vol.4, no.6, pp.43-49, 2012.
[30] S. Dey, "An Image Encryption Method: SD-Advanced Image Encryption Standard: SD-AIES", International Journal of Cyber-Security and Digital Forensics (IJCSDF), Volume 1, Issue 2, pp. 82-88, 2012.
[31] S. Dey, "Amalgamation of Cyclic Bit Operation in SD-EI Image Encryption Method: An Advanced Version of SD-EI Method: SD-EI Ver-2", International Journal of Cyber-Security and Digital Forensics (IJCSDF), Volume 1, Issue 3, pp. 221-225, 2012.

[32] S. Dey, 2014. "A beginner's guide to computer science research". ACM XRDS 20, 4 (June 2014), pp. 14-14.
[33] S. Dey, A. K. Singh, K. McDonald-Maier, "Energy Efficiency and Reliability of Computer Vision Applications on Heterogeneous Multi-Processor Systems-on-Chips (MPSoCs)", presented at Adaptive Many-Core Architectures and Systems workshop, York, UK, 13-15th June, 2018.
[34] S. Dey, J. Li, S. Cheng, "Web Scrapping: How Companies Can Use 'Big-Data' generated by Facebook?", presented at IET Data Analytics held on 5th December, 2013 at IET, London, United Kingdom.
[35] S. Dey, S. Sen, "Privacy Issues in Social Networking Websites: Is Facebook Revealing Our Social Life?". Presented at Symposium on Information Security, Securing Networks, Devices and Applications 2013 (SIS SNDA 2013) held on 15th and 16th November, 2013 at BITS Pilani, Hyderabad, India.
[36] S. Dey, 2014, "Efficient Data Input/Output (I/O) for Finite Difference Time Domain (FDTD) Computation on Graphics Processing Unit (GPU)". Available at https://www.escholar.manchester.ac.uk/item/?pid=uk-ac-man-scw:234111
[37] J. Garay, A. Kiayias, and N. Leonardos, The Bitcoin Backbone Protocol: Analysis and Applications, pp. 281–310, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.
[38] A. Gervais, G. O. Karame, V. Capkun, and S. Capkun, "Is bitcoin a decentralized currency?," IEEE Security Privacy, vol. 12, pp. 54–60, May 2014.
[39] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in 2016 IEEE Symposium on Security and Privacy (SP'16), pp. 839–858, May 2016.
[40] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," in Proceedings of ACM SIGSAC Conference on Computer and Communications Security (CCS'16), pp. 17–30, New York, NY, USA, 2016.
[41] W. T. Tsai, R. Blower, Y. Zhu, and L. Yu, "A system view of financial blockchains," in IEEE Symposium on Service-Oriented System Engineering (SOSE'16), pp. 450–457, Mar. 2016.
[42] H. Watanabe, S. Fujimura, A. Nakadaira, Y. Miyazaki, A. Akutsu, and J. Kishigami, "Blockchain contract: Securing a blockchain applied to smart contracts," in IEEE International Conference on Consumer Electronics (ICCE'16), pp. 467–468, Jan. 2016.

Authors' Profiles

Somdip Dey received his B.Sc (Hons.) in Computer Science from St. Xavier's College (Autonomous), Kolkata, India in 2012 and also represented as a Microsoft Student Partner during the period of his undergraduate studies. After completing his undergraduate studies Somdip worked as a Teaching and Research Assistant in the same institution. In 2014 Somdip completed his postgraduate study and received the degree of M.Sc. in Advanced Computer Science with specialization in Computer Systems Engineering from the University of Manchester, U.K. After his post-graduation he worked in the industry for several years and now he is pursuing his Ph.D. in Computer Science and Electronic Engineering from the University of Essex, U.K.