Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

SIMULATION BASED EVALUATION AND TUNING OF DISTRIBUTED FRAUD DETECTION ALGORITHM

Jānis Grabis

Institute of Information Technology
Riga Technical University
Kalku 1
Riga, LV-1658, LATVIA

Arturs Rasnacis

SIA TrustSearch
Generala Radzina krastmala 21 - 34
Riga, LV-1050, LATVIA

ABSTRACT

A community based fraud detection is one of the methods to ensure trustworthiness of Internet resources. The TrustSearch platform has been developed to provide community based fraud detection services. It allows Internet user to submit application reporting potential fraudulent Internet resources and relies on a consensus seeking algorithm to approve or reject the application. The system exhibits complex and dynamic behavior, and simulation is used to evaluate its performance and to determine appropriate operational parameters. The objective is to find an appropriate trade-off between evaluation accuracy and efficiency what is a characteristic challenge in distributed decision-making systems. An agent-oriented simulation model is developed and experimental studies are conducted. It has been shown that sufficiently high evaluation accuracy can be achieved and the results are remarkably robust. However, a relatively large number of participants is required. The community based platform uses blockchain technologies to reward participants for their contributions.

1 INTRODUCTION

The Internet and World Wide Web are inherently distributed systems. Unfortunately, fraudulent resources are published and distributed over the Internet. Community and crowdsourcing based fraud detection (Sauerwein et al. 2015) is one of the methods employed to detect such resources. It involves Internet users to discover and report potentially fraudulent web resources, and an agreement should be reach among the users about true nature of the reported fraud cases. This kind of approach is characterized by complex internal dynamics leading to emergent collective behavior. Simulation is a suitable technique for analyzing this behavior (Bernon et al. 2007).

This paper focuses on simulation based evaluation of a fraud detection algorithm implemented in the community based fraud detection platform TheTrustSearch.com (formerly CryptoPolice.com). Internet users submit applications reporting potential fraudulent resources in the platform. These applications are evaluated by a community of experts who have enlisted with the platform. The evaluation is performed in three stages to ensure trustworthiness of fraud identification done following a consensus building algorithm (CryptoPolice 2018). The experts are rewarded for participation in the evaluation using tokens handled by blockchain and smart contracts based technologies. The platform aims to balance decision-making accuracy and efficiency. Viability and health of the expert community is also to be maintained. Simulation is used to determine appropriate values of parameters of the decision-making algorithm and to analyze behavior of the fraud detection solution.

The objective of the paper is to develop a simulation model of the community based fraud detection algorithm and to conduct experimental studies to tune the parameters of this algorithm. An agent-based approach is used to build the simulation model. The main parameters characterizing decision-making circumstances are agents' performance in terms of decision-making accuracy and response time. The

978-1-7281-3283-9/19/$31.00 ©2019 IEEE

main control variable is the number of agents involved in decision-making at different evaluation levels. The algorithm's performance is measured in terms of accuracy, time to reach the decision and total number of agents involved in evaluation.

The rest of the paper is organized as follows. Section 2 briefly reviews research on community based fraud detection and described the decision-making algorithm used by the TrustSearch platform. The simulation model is elaborated in Section 3. The experimental studies to determine operational parameters of the algorithm are reported in Section 4. Section 5 concludes.

2 BACKGROUND

This section discusses general aspects of decentralized fraud detection, defines main requirements towards the fraud detection algorithm and describes the TrustSearch algorithm.

2.1 Fraud detection problem

The cornerstone of centralized fraud detection is an entity authorized to monitor and identify fraudulent operations. In the case of decentralized fraud detection, there is no such authorized entity and other mechanisms for fraud detection and building trust are necessary. Decentralized fraud-detection can be performed by a loose or organized community of experts. Sauerwein et al. (2015) review research work on crowdsourcing based information security. They identify that trust issues, long evaluation time frame, proper incentives, user requirements and quality of inputs are among the main concerns. Identification of phishing web sites is one of the areas where distributed decision-making is used frequently. Moore and Clayton (2007) analyze one of the most popular community based services PhishTank. They show that submission verification takes 48 hours on average. The decisions made are mostly correct though not comprehensive. Characteristics of participants strongly affect decision-making results. It has been shown that introduction of multi-stage evaluation helps to improves fraud detection accuracy (Li et al. 2017). The incentives play a major role in maintaining a viable community (Chia 2011) and blockchains have emerged as a suitable solution to provide these incentives (Cai and Zhu 2016).

The community based fraud identification systems are complex systems and simulation is an appropriate technique for specifying requirements towards these systems (Aiello et al. 2017). Simulation has been applied to study security related concerns as well. Lopez-Rojas et al. (2017) simulate introduction of fraud controls in financial transactions. An agent-based simulation approach is applied. Legato and Mazza (2017) study composition of cybersecurity teams by simulation based optimization. It is shown that team formation dynamics has a major impact on ensuring security. Recently, simulation has been applied for evaluation of security concerns in distributed systems (Lee and Wei 2016; Panagopoulos et al. 2017) though there are few works on its application in analysis of community based fraud identification.

The distributed and decentralized systems do not have the same level of trust as many systems having the centralized authority. The evaluation should be highly accurate on persistent basis because any extraordinary situation could lead to significant reduction of trustworthiness of the decentralized system. Fraud may propagate over the Internet quickly and any fraud detection service should be able to detect it as soon as possible and preferably before any adverse consequences. Finally, the decentralized system relies on participation of a loose group of experts and a viable community should have the right number of experts to balance accuracy and efficiency. To summarize, the three crucial requirements towards the decentralized fraud identification systems are:

- Evaluations should be accurate;
- Evaluation results should be obtained as fast as possible;
- The optimal number of experts should be involved.

2.2 Decision-making algorithm

The algorithm (CryptoPolice 2018) is developed by a company providing a decentralized fraud detection service. It is designed having in mind the requirements identified in the previous sub-section. The company has developed the whole distributed and decentralized fraud detection ecosystem, which includes the algorithm, development of fraud detection expert community, digital currency based rewards system and tools. This paper focuses solely on the decision-making algorithm and tuning of its parameters using simulation.

The purpose of the algorithm is to evaluate cases of fraudulent Internet resources reported by Internet users. The evaluation result is a judgment on fraudulent nature of the resources. The results are stored in a data base listing these fraudulent resources. The evaluation (Figure 1) starts with an application submitted by any Internet user. The application describes a fraudulent resource. The fraud identification platform has enlisted a number of experts called officers. The officers are responsible for evaluation of the applications and are divided in three decision-making levels referred as to L1, L2, and L3.

The officers are drawn from an open pool of internet users. Anyone can apply to become an officer. The applicants have to fill out an application form used for initial screening, undergo online training and to provide security deposit. That allows to improve decision-making quality and to reduce fraudulent applications. At the same time a blockchain based solution is used to reward the officers for their effort.

Once the application has been received it is allocated to a number of the L1 level officers. These officers issue a verdict on accepting or rejecting the application (accepting means that the officer considers that the case reported in the application is indeed fraudulent). Every verdict created is allocated to a number of L2 level officers who either accept or reject the verdict. If the L2 level officers decide to reject the verdict then a new verdict (the opposite to the L1 level) is created and added for evaluation to other L2 level officers. The decisions made at the L2 level are passed over to the L3 or final approval level. Officers at the L3 level pick up the evaluated verdicts and vote for their approval. The whole process is stopped as soon as one of the verdict evaluations receives a pre-specified amount of votes. That is represented as an interrupting event in the process diagram drawn using the BPMN notation. It is important to note that the evaluation process is dynamic and every verdict evaluation has its own evaluation thread. The next evaluation step in the thread is started as soon as the previous step has been completed (i.e., one verdict can still be at the L1 level, while another is already gathering approvals at the L2 level). Variations in decision-making timing are caused by different response rates among the officers.

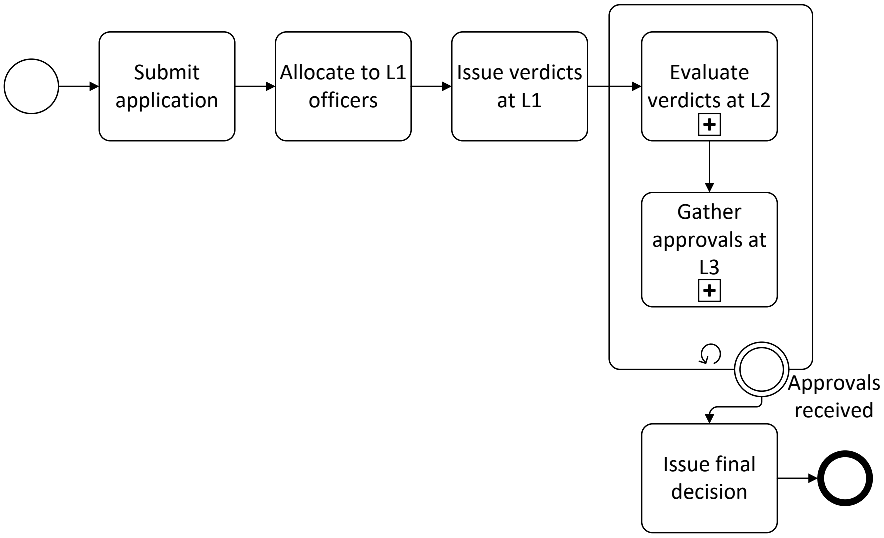

Figure 1: The community based fraud detection algorithm.

The evaluation accuracy is achieved by involving many officers and cross-checking of decisions at different levels of identification. However, many officers are necessary to maintain a viable fraud detection ecosystem. Appropriate incentives should be provided to the officers. Therefore, it is important to optimize the number of officers. The optimization seeks for a trade-off between accuracy and expenses due to involving many officers. The algorithm does not employ traditional majority voting because time to reach the final decision might be too long (additionally, more and more officers are involved if the process is longer). Rather than that competition among the verdicts is employed to gather approvals as fast as possible. A simulation model is developed to deal with the aforementioned issues and to tune parameters of the algorithm.

3 SIMULATION MODEL

The decision-making problem focuses on interactions among multiple parties. Therefore, an agent based simulation approach is chosen (MacAl and North 2010). The simulation model is developed using a general purpose programming language. Three pools of agents representing officers at different level of decision-making are created (Figure 2). The agents are characterized by their ability to make a correct judgment. Their level of experience and inclination towards malicious intent also could be modeled. It is assumed that all officers perform their activities independently. The decision-time is tracked.

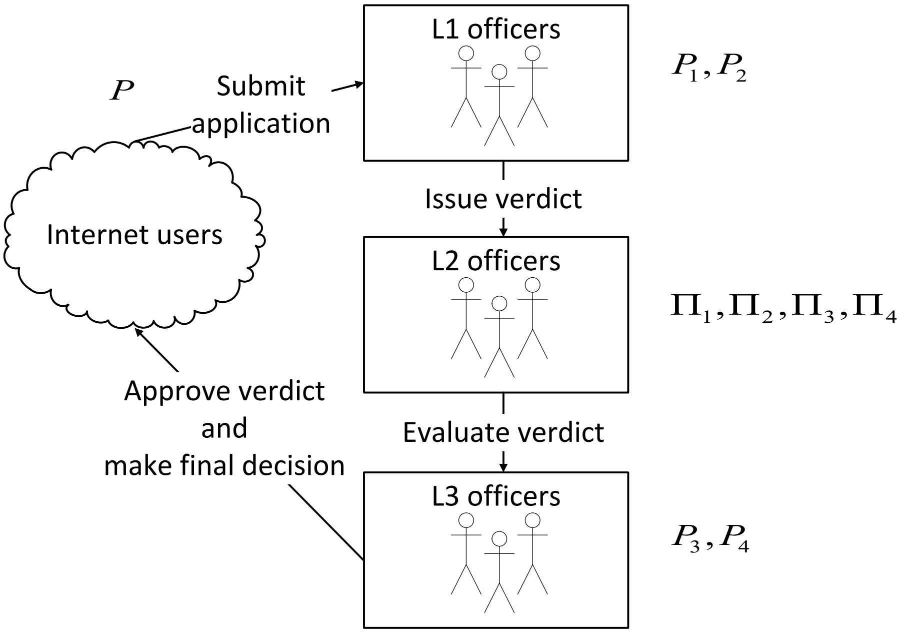

Figure 2: Interactions among agents in the simulation model.

At the beginning of the simulation, agents representing officers at all three levels are created and their attribute values are initialized. The number of required approvals by L3 officers *S* is also set. The simulation process is as follows:

1. An application is generated. It is either correct with probability *P* or false with probability 1- *P*.
2. N_1 officers are randomly drawn from the pool of L1 officers.
3. Every officer issues a verdict v_i .

   a. The probability that the verdict accepts a correct application is P_1 (accordingly $1-P_1$ is the probability of rejecting a correct application) and the probability that the verdict accepts a wrong application is P_2 .
   b. It takes time $t_i^1 \sim LogNormal(\mu_1,\sigma_1)$ to issue the *i*th verdict μ_1 is the average time to issue the verdict and σ_1 is the standard deviation. The lognormal distribution is used to represent the fact that most of the officers take up and complete the task quickly and there is a small number of officers taking long time to issue the verdict.
4. Every verdict is evaluated by N_2 officers randomly drawn from the pool of L2 officers.
5. The evaluation result w_{ij} (index *i* refers to the verdict and index *j* refers to the L2 officer performing the evaluation) is generated according to conditional probabilities Π_j defined in the experimental design.
   a. It takes time $t_{ij}^2 \sim LogNormal(\mu_2,\sigma_2)$ to complete the verdict evaluation. μ_2 is the average verdict evaluation time and σ_2 is the standard deviation.
   b. The number of approvals for w_{ij} is set $M_{ij}=0$.
6. The verdict evaluations w_{ij} are passed over to L3 officers. The probability that an L3 officer makes a decision at a given time period is P_3 and the probability that the decision is to approve the verdict is P_4 .
   a. If verdict is approved then $M_{ij}=M_{ij}+1$.
7. The process (Step 3 to Step 6) is stopped as soon as there is a verdict evaluation $w_{i^*j^*}$ (*i** refers to the verdict and j^* refers to the L2 officer performing the evaluation first to receive the required number of L3 level approvals), which has received *S* approvals at the L3 level (i.e., $M_{i^*j^*}=S$).
8. The decision-making time is evaluated $T=t_{i^*}^1+t_{i^*j^*}^2+t_3$, where t_3 is the time period between $w_{i^*j^*}$ entering the L3 level evaluation and receiving *S* approvals.
9. The process is replicated *R* times starting with Step 1.

The simulation model is used to evaluate decision making accuracy and time depending on process parameters.

4 EXPERIMENTAL STUDIES

The objective of the experimental studies is to fine-tune the proposed fraud detection algorithm to balance trustworthiness and efficiency of the crowdsourced decision-making. The two main performance measures are:

1. Accuracy of fraud identification – what is the ratio of correctly evaluated applications to all applications? An application evaluation is correct if officers approve a valid application or reject a false application;
2. Evaluation time – how long does it take to evaluate an application what is measured in a number of time periods?

The main control variable is the number of officers involved in the evaluations. Thus, the research question is what is an appropriate number of officers involved in fraud detection?

4.1 Experimental Design

The fraud detection platform is in early testing stages and currently there are no statistical data accumulated. Therefore, parameters of the algorithm and attributes of officers are selected from a range of plausible values as suggested by experts (Table 1). It is assumed that 80% of all applications are correct what is in line with similar investigations on crowdsourced fraud detection. The probability to accept a correct application by L1 officers P_1 is has low value 0.6 and high value 0.9 and the probability to accept a wrong application by L1 officers P_2 is varied from 0.4 to 0.1. The probability of decisions made by L2 officers is conditional on evaluation done by the L1 officers. It is reasoned that identification of errors made by the L1 officers is more challenging than simply confirming their results, especially, if the initial application is wrong. Thus, $\Pi_4 < \Pi_1$ or the probability to maintain (approve) the reject verdict for a wrong application is smaller than the probability to approve the accept verdict for a correct application.

If Table 1 lists only one value, a parameter is kept constant for all experiments. A full factorial design is created by combining low and high value of all parameters having the levels specified. As the result, 256 experimental treatments are considered. Two hundred replications were performed for every treatment. A few additional experiments are also conducted using extreme value of parameters for stress testing purposes.

Table 1: Experimental value of the fraud detection algorithm.

Parameter	**Value**	**Definition**
N_1	5;20	Number of officers involved in evaluation at the L1 level
N_2	5;20	Number of officers involved in evaluation at the L2 level
P	0.8	Probability of application being correct
P_1	0.6; 0.9	Probability that the verdict approves the correct application
P_2	0.4;0.1	Probability that the verdict approves the wrong application
Π_1	0.7;0.9	Probability that L2 level officer approves the accept verdict for the correct application (correct decision)
Π_2	0.1;0.2	Probability that L2 level officer declines the accept verdict for the correct application (wrong decision)
Π_3	0.1;0.3	Probability that L2 level officer declines the reject verdict for wrong the application (wrong decision)
Π_4	0.5;0.7	Probability that L2 level officer approves the reject verdict for the wrong application (correct decision)
P_3	0.02	Probability to make a decision by L3 level officer in any given time period
P_4	0.95	Probability of L3 level officer confirming the correct L2 decision
t_i^1	$LogNormal(5,2)$	Time to issue a verdict time at L1
t_{ij}^2	$LogNormal(10,5)$	Time to issue a verdict evaluation at L2
S	6	Number of approvals required at the L3 level

The simulation model is implemented using a general purpose programming language Python. Separate threads are used to represent each evaluation chain staring from the application to the approval. The simulation was performed on a quad-core personal computer with 3.6 GHz CPU and 8 GB RAM. In the case N_1=5 and N_2=5, the execution time of a single experimental treatment was about 105 seconds. In the case N_1=20 and N_2=20, the execution time of a single experimental treatment was about 522 seconds and the memory usage was 300 MB.

4.2 Results

The first set of experimental results focuses on selection of appropriate number of officers at levels L1 and L2. Table 2 and Figure 3 show the average share of correct decisions depending on the number of officers. The table indicates that the best accuracy achieved is 95% what was deemed as acceptable by stakeholders of the fraud detection solution and it is higher than accuracy of initial applications. The figure also shows intermediate values of N_1 suggesting that changes in accuracy are steady. Increasing the number of officers at the L1 levels helps improving the accuracy. Moreover, this number should not be smaller than the number of officers at the L2 level. If there are more L2 officers than L1 officers then it possible that errors made at the L1 level propagate through the system. The total number of officers involved in the decision-making process is also calculated. Its average value for all treatments is 6793. The stakeholders accepted this number though it is relatively high since reward tokens are issued only to those officers who have participated in the winning thread of evaluation. Surprisingly, the accuracy was not significantly affected by P_1 (having levels of 0.6 and 0.9) indicating that errors made at the L1 level can be easily correct at the L2 level.

Table 2: The accuracy depending on number of officers at levels L1 and L2

Number of officers at L1 (N_1)	Number of officers at L2 (N_2)	Accuracy
5	5	0.929
20	**5**	**0.955**
5	20	0.876
20	20	0.881

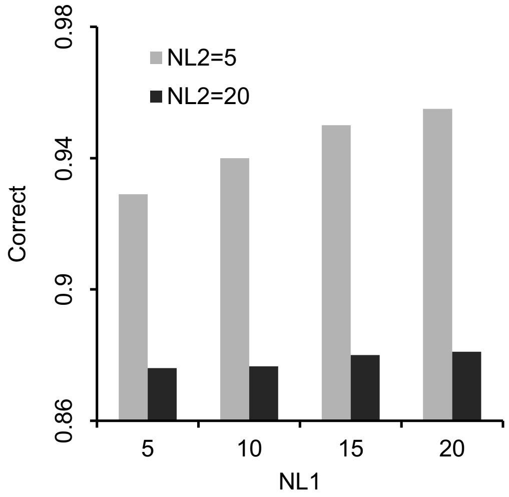

Figure 3: The application evaluation accuracy (Correct) depending on the number of officers N_1 (NL1) and N_2 (NL2) at levels L1 and L2, respectively.

According to the linear model fitted, the fraud detection accuracy is fairly robust with regards to quality of decisions made at the L1 level. The impact of officers' performance at the L2 level is also evaluated (Figure 4). The results show that the probability of making right decisions at the L2 level significantly affects decision-making accuracy. Π_1 is of particular importance partially because correct verdicts occur more often than wrong ones. This observation also implies that the officers might be nudged towards rejecting initial verdicts.

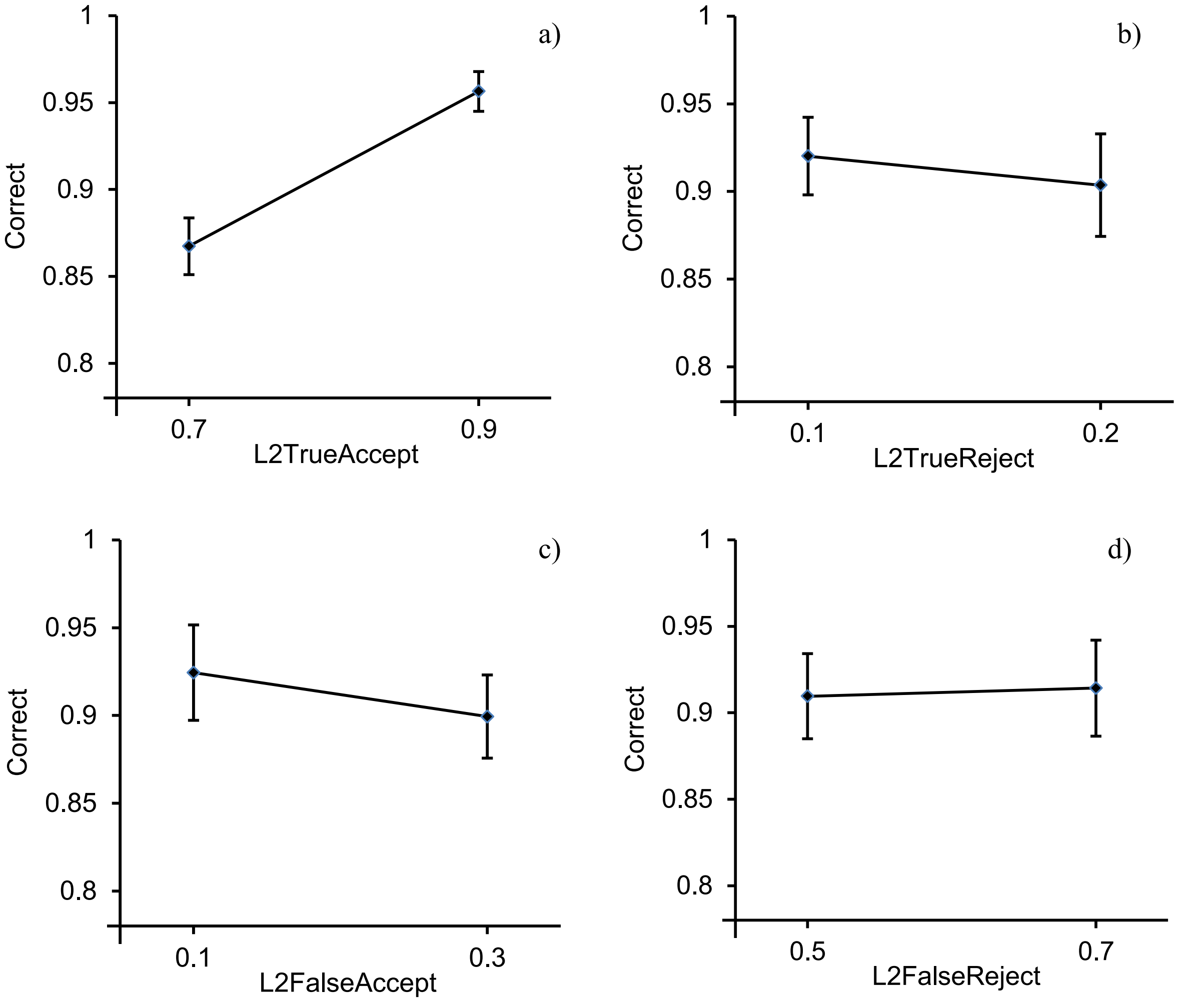

Figure 4: The accuracy of final evaluation depending on probability that: a) L2 level officer approves correct verdict (Π_1; L2TrueAccept); b) L2 level officer declines correct verdict (Π_2; L2TrueReject); c) L2 level officer approves wrong verdict (Π_3; L2FalseAccept); and d) L2 level officer declines wrong verdict (i.e., L2 decision is correct (Π_4;L2FalseReject).

The evaluation time is another important aspect and decisions should be made as promptly as possible. Figure 5 shows that the increasing the number of officers at level L1 significantly reduces the application evaluation time. There are more evaluation threads and chances increase that one of them will be completed faster than with fewer officers. However, the average number of officers involved increases from 3 480 to 10 106 and in practice there might be difficulties to attract such a number of equally dedicated contributors. As noted before, the increasing number of officers also leads to accuracy improvements.

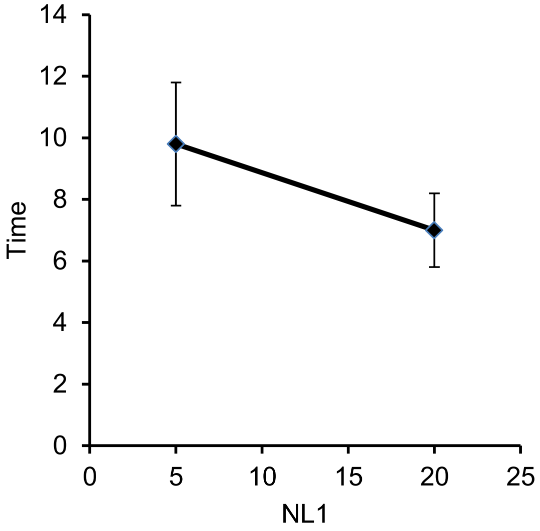

Figure 5: Evaluation time in abstract units depending on the number of officers N_1 at level L1.

The simulation results indicate that the fraud detection algorithm is quite robust. Therefore, additional evaluation scenarios are considered (Table 3). In these scenarios, it is assumed that officers severely underperform what might occur due to lack of knowledge, motivation or good intentions. Scenario S10 represents a situation when almost all officers make wrong decisions at all levels of decision making and a near zero accuracy is an obvious result. However, as officers' performance increases above 50% the accuracy quickly approaches 78% for the scenario S70. This accuracy is still not acceptable to the stakeholder though such a low officers' performance is not likely in practice. The results indicated that the algorithm is quite resilient even in very adverse situations.

Table 3: The fraud identification accuracy in the case of underperforming officers.

Scenario	**Parameters**	**Accuracy**
S10	$P_1 = 0.1,\ \Pi_1 = 0.1, \Pi_2 = 0.9, \Pi_3 = 0.9, \Pi_4 = 0.1,\ P_4 = 0.1$	0.04
S50	$P_1 = 0.5,\ \Pi_1 = 0.5, \Pi_2 = 0.5,\ \Pi_3 = 0.5, \Pi_4 = 0.5,\ P_4 = 0.5$	0.49
S60	$P_1 = 0.6,\ \Pi_1 = 0.6, \Pi_2 = 0.4,\ \Pi_3 = 0.4,\ \Pi_4 = 0.6, P_4 = 0.6$	0.61
S70	$P_1 = 0.6,\ \Pi_1 = 0.6, \Pi_2 = 0.3,\ \Pi_3 = 0.3, \Pi_4 = 0.7, P_4 = 0.7$	0.78

5 CONCLUSION

The paper focuses on fine tuning of the TrustSearch's community based fraud detection algorithm. It provides practical contribution as well as demonstrates a novel application of simulation in evaluation of community based fraud detection. The experimental evaluation shows that the algorithm potentially gives satisfactory fraud detection accuracy and the number of officers can be varied effectively to improve the accuracy. Both accuracy and decision-making time can be improved by increasing the number of officers, especially, at the L1 level. However, that increases the number of officers involved exponentially what might have negative consequences on viability of the community and devaluate tokens issued to motivate the officers. The stakeholders consider the simulation model as a valuable tool to demonstrate capabilities and limitations of the crowdsourced fraud detection platform.

The identified trade-off between accuracy and efficiency is consistent with previous findings on distributed decision-making, and the simulation model developed allows to search for a suitable balance between these two contradicting objectives. The algorithm only considers application submitted by Internet users and it cannot estimate comprehensiveness of the proposed fraud detection platform.

The main limitations of the current model are that the model is evaluated using parameters suggested by the experts and all officers are treated as having the same overall characteristics like evaluation and accuracy and response time. The current results provide insight concerning trends and interrelationships among parameters while evaluation of actual values of performance measurements requires additional input data and validation. The simulation model is computationally capable to deal with the networks of the size considered (especially since business-wise the increase of the number of officers is not desirable). However, if every application is not treated individually or network capacity should be analyzed then refactoring of the model to reduce consumption of computational resources will be needed.

The current investigation was aimed to characterize the overall behavior of the systems. It is decided that more detailed fine-tuning and optimization of decision-making time will be possible when actual data are gathered. More specifically, the stakeholders are interested to consider various attributes of officers such as experience and intent.

REFERENCES

Aiello, F., A. Garro, Y. Lemmens, and A. Dutre. 2017. "Simulation-based Verification of System Requirements: An Integrated Solution". In *Proceedings of the 2017 IEEE 14th International Conference on Networking, Sensing and Control, ICNSC 2017*, May 16th-18th, Calabria; Italy; 726-731.

Bernon, C., M. Gleizes, and G. Picard. 2007." Enhancing Self-organising Emergent Systems Design with Simulation". In *Proceedings 7th International Workshop on Engineering Societies in the Agents World, ESAW 2006*, September 6th-8th, Dublin; Ireland, 284-299.

Cai, Y. and D. Zhu. 2016. "Fraud Detections for Online Businesses: A Perspective from Blockchain Technology". *Financial Innovation* 2(1):1-10.

Chia, P.H. 2011. "Analyzing the Incentives in Community-based Security Systems". In *2011 IEEE International Conference on Pervasive Computing and Communications Workshops*, March 21st-25th, Seattle, WA, United States, 270-275.

CryptoPolice. 2018. Community Base Scam Identification Protocol : White paper. https://www.cryptopolice.com/CryptoPolice_whitepaper.pdf , accessed 10th April, 2019.

Lee, V. and H. Wei. 2016. "Exploratory Simulation Models for Fraudulent Detection in Bitcoin System". In *Proceedings of the 2016 IEEE 11th Conference on Industrial Electronics and Applications*, June 5th-7th, Hefei, China, 1972-1977.

Legato, P. and R.M. Mazza. 2016. "A Simulation Optimisation-based Approach for Team Building in Cyber Security". *International Journal of Simulation and Process Modelling* 11(6):430-442.

Li, J. and S. Wang. 2018. "PhishBox: An approach for phishing validation and detection". In *Proceedings 2017 IEEE 15th International Conference on Dependable, Autonomic and Secure Computing, 2017 IEEE 15th International Conference on Pervasive Intelligence and Computing, 2017 IEEE 3rd International Conference on Big Data Intelligence and Computing and 2017 IEEE Cyber Science and Technology Congress*, November 6th-11th, Orlando, FL, United States, 557-564.

Lopez-Rojas, E.A., Axelsson, S., and D. Baca. 2018. "Analysis of fraud controls using the PaySim financial simulator". *International Journal of Simulation and Process Modelling* 13(4):377-386.

MacAl, C.M. and M.J. North. 2010. "Tutorial on agent-based modelling and simulation". *Journal of Simulation*. 4(3):151-162.

Moore, T. and R. Clayton. 2008. "Evaluating the wisdom of crowds in assessing phishing Websites". In *12th International Conference on Financial Cryptography and Data Security*, January 28th-31st, Cozumel, Mexico, 16-30.

Panagopoulos, A., Koutrouli, E., and A. Tsalgatidou. 2017. "Modeling and evaluating a robust feedback-based reputation system for e-commerce platforms". *ACM Transactions on the Web* 11(3):1-55.

Sauerwein, C., Gander, M., Felderer, M., and R. Breu. 2016. "A systematic literature review of crowdsourcing-based research in information security". In *Proceedings 2016 IEEE Symposium on Service-Oriented System Engineering*, March 29th-April 1st, Oxford; United Kingdom, 364-371.

Singh, A.M. and S. Jatinder. 2018. "Hybrid optimization algorithm for community and fraud detection in complex networks for high immunity towards link and node failures". *International Journal of Intelligent Engineering and Systems* 11(1):211-220.

AUTHOR BIOGRAPHIES

JĀNIS GRABIS is a Professor at the Faculty of Computer Science and Information Technology, Riga Technical University, Latvia. He obtained his PhD from the Riga Technical University in 2001 and worked as a Research Associate at the College of Engineering and Computer Science, University of Michigan-Dearborn. He has published in major academic journals including OMEGA, European Journal of Operational Management, International Journal of Production Research, Computers & Industrial

Engineering and others. He has been a guest-editor for two top academic journals and member of the program committee of several academic conferences. Janis Grabis has co-authored a monograph on supply chain configuration published by Springer. His research interests are in supply chain management, enterprise applications and project management. His email address is grabis@iti.rtu.lv.

ARTURS RASNACIS is CEO at TrustSearch. He obtained the Master degree in Information Technology from the Riga Technical University in 2016. His research and professional interests are blockchains, cryptocurrencies and agile development. He has published research papers on team dynamics in agile development. His email address is art@cunami.lv.