PAYSIM: A FINANCIAL MOBILE MONEY SIMULATOR FOR FRAUD DETECTION

Edgar Alonso Lopez-Rojas[a], **Ahmad Elmir**[b], and **Stefan Axelsson**[c]

[a],[b] Blekinge Institute of Technology , [c] The Norwegian University of Science and Technology
[a] edgar.lopez@bth.se [b] ahel11@bth.se [c] stefan.axelsson@hig.no

ABSTRACT

The lack of legitimate datasets on mobile money transactions to perform research on in the domain of fraud detection is a big problem today in the scientific community. Part of the problem is the intrinsic private nature of financial transactions, that leads to no public available data sets.

This will leave the researchers with the burden of first harnessing the dataset before performing the actual research on it. This paper propose an approach to such a problem that we named the PaySim simulator.

PaySim is a financial simulator that simulates mobile money transactions based on an original dataset. In this paper, we present a solution to ultimately yield the possibility to simulate mobile money transactions in such a way that they become similar to the original dataset. With technology frameworks such as Agent-Based simulation techniques, and the application of mathematical statistics, we show in this paper that the simulated data can be as prudent as the original dataset for research.

Keywords: Multi-Agent Based Simulation, Financial data, Fraud Detection,Retail Fraud, Synthetic Data.

1. INTRODUCTION

Obtaining access to data sets of mobile transactions for research is a very hard task due to the intrinsic private nature of such transactions. Scientists and researchers must today spend time and effort in obtaining permits and access to relevant data sets before they can research on such data set. This is time consuming and distracts researchers from from focusing on the main problem which is performing experiments on the data and finding novel ways to solve problems such as the problem that inspired this paper which is the fraud detection on financial data.

The work presented in this paper provides a tool and a method to generate synthetic data with the help of a simulator that we named *PaySim*. PaySim generates synthetic datasets similar to real datasets from mobile money transactions. This will be done by the means of computer simulation, in particular, agent based simulation. Agent based simulation is of great benefit in this context, this partially because the models created represent with accuracy the human behaviour during transactions and are flexible enough to easily be adapted to new constraints.

PaySim simulates mobile money transactions based on a sample of real transactions extracted from the logs of a mobile money service implemented in an African country. With the help of statistic analysis and social network analysis PaySim is able to generate congruous results with the original data set.

The scope of this paper covers the design and construction of the simulator as well as the evaluation of the quality of the data generated. The injection of malicious fraud behaviour and the application of different fraud detection methods are outside the scope of this paper and are the topics for further work with the PaySim simulator.

Outline This paper is structured as follows: Section 2. presents the background and previous work in simulating financial data. Section 3. states the problem and during sections 4. and 5. we present the implementation of PaySim and the results of the simulations.Finally section 6. present the conclusions and future work.

2. BACKGROUND AND PREVIOUS WORK

In many parts of Africa the adoption of mobile money as a means of sending & receiving funds have improved the life of merchants and customers alike. In Tanzania for instance, which according to the world bank is one of the fastest growing economies in the world, the adoption of mobile money as a solution for creating payments has induced a positive effect on the overall economy. During December 2013 alone, 100 million transactions were made in total netting a volume of $1.8 billion dollars (Seetharam and Johnson, 2015).

The domain of Mobile Money Transfer has grown substantially in the last few years and have attracted greater attention from users, specifically in areas in which banking solutions may not be as procurable as in developed countries. Many solutions have been employed in many places for this purpose. There are existing mobile money services in more than 10 African countries which coverage of 14% of all mobile subscribers (Rieke et al., 2013).

The ever growing usage of mobile money has increased the chances and likelihood of criminals to perform fraudulent activities in an attempt to circumvent the security measures of mobile money transfers services for personal financial gain. There is therefore a great amount of pressure on researching the potential security pitfalls that can be exploited with the ultimate goal to develop countersolutions for the attacks.

Due to the large amount of transactions and the ever changing characteristics on fraud, the current measures against fraud lack effectiveness. Many current system still base their detection mechanism on simple thresholds

assigned arbitrarily. Therefore there is a need to push forward and investigate the effect of fraud and stop the wrongdoers from fraudulent profit.

With *PaySim*, we aim to address this problem by providing a simulation tool and a method to generate synthetic datasets of mobile transactions. The benefits of using a simulator to address fraud detection was first presented by (Lopez-Rojas and Axelsson, 2012b). This research states the problem of obtaining access to financial datasets and propose using synthetic datasets based on simulations. The method proposed is based on the concept of MABS (Multi Agent Based Simulation). MABS has the benefits that allows the agents to incorporate similar financial behaviour to the one present in domains such as bank transactions and mobile payments.

The first implementation of a simulator for financial transaction was introduced by (Lopez-Rojas and Axelsson, 2012a) with a mobile money transactions simulator. This simulator was implemented due to the difficulties to implement a proper fraud detection control on a mobile money system that was under development. This paper was the first to present an alternative to the lack of real data problem. The synthetic dataset generated by the simulator was used to test the performance of different machine learning algorithms in finding patterns of money laundering.

The work by (Gaber et al., 2013) introduced another similar technique to generate synthetic logs for fraud detection. The main difference here was that this time there was available real data to calibrate the results and compare the quality of the result of the simulator. The purpose of this study was to generate testing data that researchers can use to evaluate different approaches. This works differers significantly from our work because we present a different method for analysing the data place special attention on evaluating the quality of the resultant synthetic data set.

There has been some work done in the domain of financial transactions for retail stores. The most prominent of which is the work done by (Lopez-Rojas et al., 2013). The work done in that paper is very similar to the work done in this paper. A large collection of data was gathered from Sweden's biggest shoe-retailer, and techniques involved complex machine-learning algorithms in an attempt to find fraudulent behaviour in clients. The paper showed among other things results from Social Network which described the relationship between the clients and the sellers for each store. A definition of what was perceived as "fraudulent" was made and based on that the machine-learning algorithms were trained to detect that type of behaviour.

Public databases of financial transactions are almost non existent. However the work of (Lopez-Rojas and Axelsson, 2014) during the implementation of a simulator called BankSim presents a MABS of financial payments. BankSim is implemented in a similar way as the RetSim simulator and our simulator using in addition to statistical analysis a social network analysis. BankSim is based on the aggregated financial information of payments during 6 months of the two main cities of Spain that was provided by a bank in Spain with the purpose of developing applications of different kinds that benefit from this sort of data. Our work differs from this work because the source of the data and the characteristics of bank payments and mobile transactions are different as presented later in the following sections.

The key common aspect on previous work is the use of the paradigm of "Multi Agent Based Simulation" approach which incorporates into the behaviour of the agents the main customer logic to reach similar results as the real world. It is important to recognize that a simulation is not an actual "replication" of the original data set. Rather, a simulation will with the aid of statistical methods generate a very similar data set of the original data set. The degree in variance will largely be dependent on how the data on the original data set is structured, hence, different simulations based on different seeds will generate different output data sets but consistent with the real world.

3. PROBLEM

The problem formulation for this research paper tackles the issue of whether the generation of synthetic financial data is sufficient to supersede real financial data whilst simultaneously yield commensurable results if the synthetic data is used as the source data set for any research. This is of primary concern for any researcher that wish to perform scientific tests but does not or have limited access to a real financial data set.

The main focus and goal for the simulation is to yield another completely self-sufficient data set with the goal of having similar statistical properties as the original data set. To yield such results, the simulator must go through several steps to be able to complete.

In order to simulate the mobile money service, we need to properly simulate the different kind of transactions that the system supports. We decided to cover 5 of the most important transaction types: CASH-IN, CASH-OUT, DEBIT, PAYMENT and TRANSFER.

CASH-IN is the process of increasing the balance of account by paying in cash to a merchant.

CASH-OUT is the opposite process of CASH-IN, it means to withdraw cash from a merchant which decreases the balance of the account.

DEBIT is similar process than CASH-OUT and involves sending the money from the mobile money service to a bank account.

PAYMENT is the process of paying for goods or services to merchants which decreases the balance of the account and increases the balance of the receiver.

TRANSFER is the process of sending money to another user of the service through the mobile money platform.

There are other types of transactions that we decided to exclude from the simulation due to the low percentage of data found in the sample.

4. MODEL AND IMPLEMENTATION

PaySim uses the MABS toolkit called MASON version 19 which is implemented in Java (Luke, 2005). We selected MASON because it is: multi-platform, supports parallelisation, and fast execution speed in comparison with other agent frameworks. This is especially important for multiple running and computationally expensive simulations such as PaySim (Railsback et al., 2006).

4.1. Overview,Design and Details (ODD)

The design of PaySim was based on the ODD model introduced by (Grimm et al., 2006). ODD contains 3 main parts: *Overview*, *Design Concepts* and *Details*.

4.1.1. ODD Overview

The purpose of this simulator is to simulate payments done in the realms of mobile transactions. The simulator should ultimately perform simulations in such a way that synthetic data in regards to mobile transactions can be generated. The simulator should generate synthetic data that is very similar to a batch of real transactional Data provided by Ericsson. The goal is to have a generator that can generate data on the fly that can later be used by the scientific community in an attempt to research more about fraud detection.

The model has one primary type of Entity which is *Client*. Each client has a profile that describes the allowed behaviour for the client such as the limit on transactions daily/yearly, the transaction limit and the maximum balance for the client. Furthermore the number of transactions, withdrawals, transfers and deposits is stored for each client. The client can further be classified by age to be young, adult or senior. Each client has a base currency in which the transactions are based upon. The client can perform transactions in the form of deposits, withdrawals and transfers. For every transaction that is made, it is stored and saved within the client.

4.1.2. Process Overview and Scheduling

The client has several processes that alter their internal states. For each step that is made by the simulator, based on a random variable that is contingent on calculated probabilities, a type of transaction that is to be performed by the client is chosen. A deposit transaction will increase the balance of the client, a withdrawal will decrease the balance of the client and a transfer transaction will withdraw money from the original client and then deposit them to the destination client in question.

The concepts that are behind the model are based on statistical analysis of a large batch of real data. From this batch of data, probabilities of each action were calculated and incorporated into the model to generate synthetic information as close as possible to the real data. The client agent has some adaptive behaviours that will alter their way of acting; for instance if the client has reached its daily limit it cannot withdraw money any more for that day. This adaptive behaviour is a direct result of the *transfer* process mentioned above. There is interaction between agents since there is a probability that at a particular step of the simulation, an agent might transfer money to another agent and thus alter its and the other agents state.

4.2. Inputs

There are multiple inputs required in order for the simulator to function smoothly. As initial input, the number of clients neighbours for each agent is assigned. The profile for each agent is then further attached based on a probability. Their location on the spatial space along with their neighbours is also initialized.

- **Parameter File** This is the file that contains all of the needed parameters that the simulator needs to initiate. Among these parameters we find the seed and perhaps the most relevant of which is the paths for where the input files and the output files are placed on the current machine.
- **Aggregated Transaction File** This file contains the distribution of the transactions from the original data set. More precisely, it contains how many transactions were made at any given day/hour combination (step). what is the average price for that, what type of transaction it was etc. This is of paramount importance for the simulator since statistical data is generated from the information gathered from this file.
- **Repetitions File** This file contains the frequency of transactions that the original clients had per type of transaction. This means that some of the agents are schedule more than others based on a social network analysis of the indegree and outdegree of the customers.

Since the simulator is using MASON as the framework for performing the simulation, it is of paramount importance to define how each step is to be regarded. For this simulation we defined that each day/hour combination represents one step. At each step, a Client that represents the agent for the simulator is generated. The client will be placed in an environment in which it is to make decisions based on the information it perceives. The Client is created with the statistical distribution of the possibilities to perform each transaction type for a specific day/hour combination. The client then randomly perform (based on the distribution initiated) different transaction types in relation to the other clients on the simulator. Also, for each client generated, there is a probability **P** for the client to make future transactions at later steps. This probability is gathered from the database of the original data set.

4.3. Initiation Stage

In this stage, the PaySim simulator must load the necessary data needed from the original dataset:

- **Load The Parameters** The first and most important step is to load the values for each parameter in the

parameter file. These will among other things contain the file paths for the source data inputs that the simulator needs to load.

- **Load Aggregate File** This is the original Aggregate File that will be used as a base point for the simulator to generate statistically similar results in terms of ”What to simulate, at what day, at which amount” etc. One such extraction could be for instance: At Day 1 and hour 15, simulate 8703 transactions of the type PAYMENT with the average transaction size of 180000 and the standard deviation of 15000.

- **Load Initial Balance Container** Apart from the statistical distribution for each transaction type input to the client, there is another important input, namely, the ”Initial Balance” of the client. Upon the generation of each client in the simulation, there must be an initial ”Balance” attached to that client. This balance is generated with the ”Balance Container” file as base point. The Balance Container consists of the different probabilities that will generate different initial balance ranges.

- **Load Maximum Repetitions File** As mentioned previously, each client has a probability P of making future transactions in future steps in the simulation. What this file does is to make sure that each client does not make more repetitions that is allowed. Like the Balance Container, the probabilities in this file is also yielded from the database of the original data set.

4.4. Execution Stage

Upon completion of the Initiation Stage and all of the parameters are successfully loaded, the simulator can now proceed to the execution stage. It is at this stage that the simulator will perform the actual simulation, and yield the simulated transaction results:

4.4.1. Generating The Clients

The agents are the founding blocks of the ”Agent Based Simulator”. The agent in this context, resembles the ”Client”. Upon each step of the simulation, the PaySim simulator will convert each step to a ”Day/hour” combination. This will then be used as an input to extract the statistical distributions from the original data set. Based on the *Aggregated Transaction File*, PaySim harness the probability **P** of performing each each transaction in the simulator and save it into the model of the client. With this information, the client now has gained more knowledge and will know the following important things:

- **Number Of Transactions** This is the total number of transactions that this generated client will do.

- **Make Future Steps** This is the information of whether the client is to participate in future steps. Which means scheduling the tasks of performing more transactions during further steps.

- **Statistical Distribution** This is the different probabilities that the client will have loaded into it which entails the probability **P** of performing each action.

- **Initial Balance** This will be the initial balance that the client will have once generated.

4.4.2. Performing the transactions

After each client is generated, the client will make the decision of what type of transaction it will ultimately make, again this is completely derived from the distribution loaded. The client is in an environment which allows it to freely interact with other clients in the simulation. There are some types of transaction types that is based on that, like ”TRANSFER” for instance. The ”TRANSFER” type is exchange of money from one client to another; hence, the client will have to interact with other clients to simulate the actual exchange of funds.

4.5. Finalization Stage

After each of the clients have completed their role in the simulation and performed all of the transactions allotted the results must be saved. There are 4 outputs generated for each simulation made. All of which serve a specific purpose which will allow for the exact repetition of the simulation with the exact initial properties.

- **Logfile** Each transaction that is made will contain a record with the meta-data for that transaction. Data such as what client performed which action, to which other client, the sum of the transaction, and the delta in balance for all clients involved. Each such record will be saved in a logfile unique for the specific simulation.

- **MySQL Database** Apart from the logfile, the record for each transaction will also be saved into a MySQL database. The purpose of which is to allow for easier queries when the analysis of the resulsts is to be made.

- **Aggregate Dump** An aggregatedump that is similar to the original aggregatedump from the original data set will also be generated. It is these two files that will be used to generate the plots and graphs resembling the results of the transactions.

- **Parameter File History** This file will contain the exact properties needed for the simulation to be able to reproduce the exact same results again. This is important because each simulator must be able to be reproduced again, and without the original ”seed” used, it will not be possible.

5. RESULTS

We ran PaySim several times using random seeds for 744 steps, representing one month of real time data. Each run took around 30 minutes in a i7 intel processor. We selected a datasets that contained the lowest difference in

Figure 1: Visualization of transaction type CASH-IN

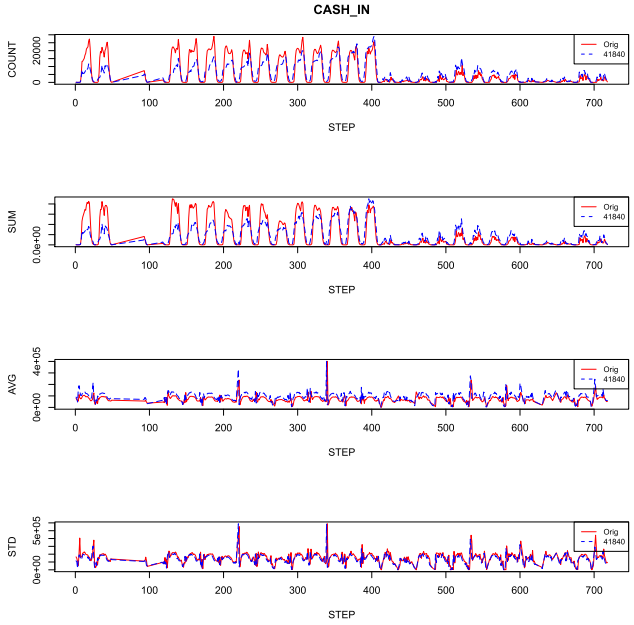


Figure 2: Visualization of transaction type CASH-OUT

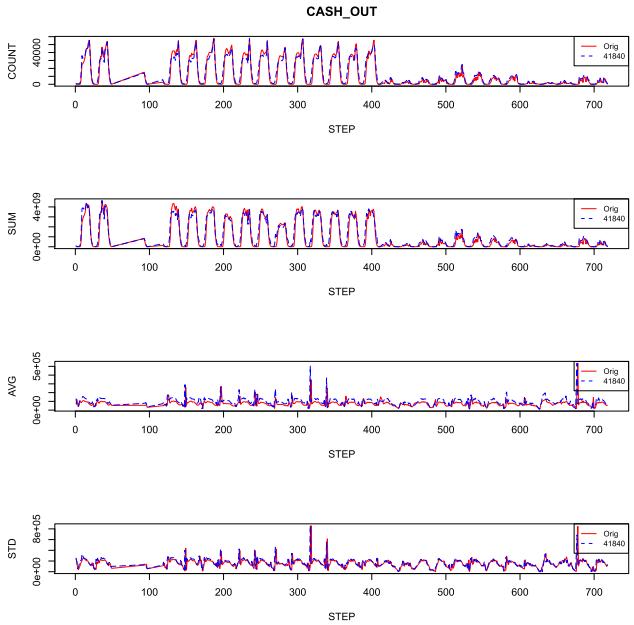


Figure 3: Visualization of transaction type TRANSFER

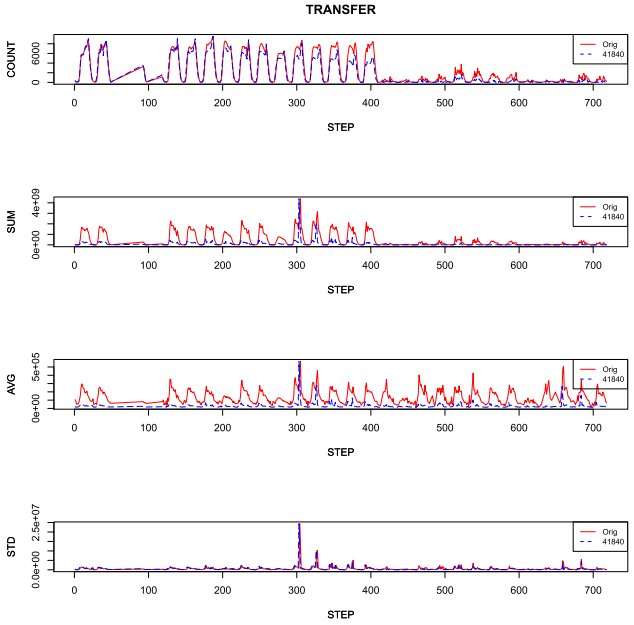


Figure 4: Visualization of transaction type PAYMENT

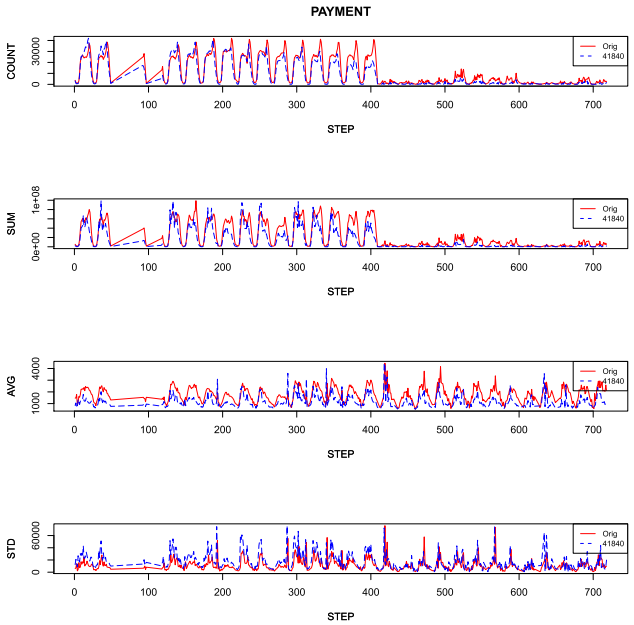
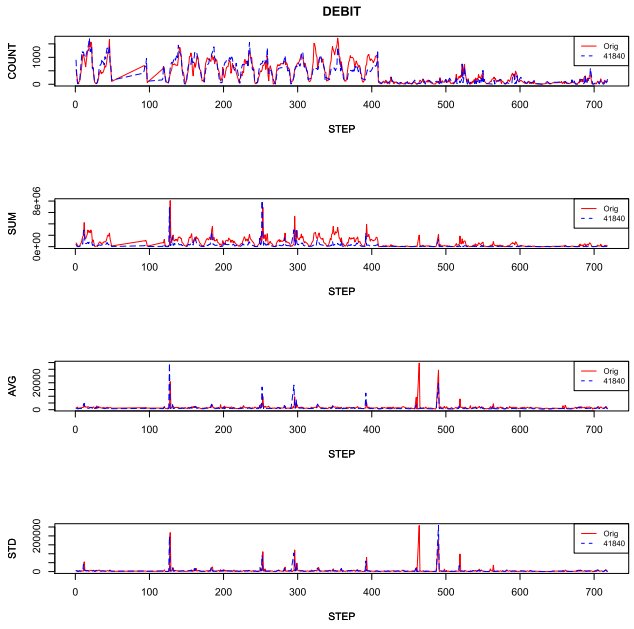


Figure 5: Visualization of transaction type DEBIT



values according to the original data set provided. The selected synthetic dataset was arbitrary named *PS41840*. *PS41840* contains around 23 million records divided into the 5 types of categories presented before. Table 1 shows the types of transactions, count and average amount generated with the simulator. The amount values are given in a currency that we can not disclose.

Table 1: Simulated PS41840

TYPE	Count	avgAmount
CASH-IN	4 496 947	153 019
CASH-OUT	9 014 407	155 989
TRANSFER	2 030 969	630 810
PAYMENT	8 955 794	10 793
DEBIT	139 935	5 016

The evaluation of the quality of the database was first calculated using the sum of square error (SSE) method on the quantities of the different datasets. The one with the lowest error was *PS41840*.

In order to verify that the simulation was working properly we plotted the distributions to visually identify significant differences between the original and the synthetic dataset. Figures 1, 2, 3, 4 and 5 show the visualization per type of transaction. Each figure contains the output for each step regarding the count of transactions, the total sum of transaction, the average and the standard deviation. The red continuous line represent the original data distribution and the blue dashed line represent the synthetic dataset *PS41840*.

Something we noted is that the first 14 days of the simulation the activity in the system is higher compared to the remaining days. This is perhaps a phenomenon present due to the introduction of income during the first days of the month.

6. CONCLUSIONS

PaySim is a simulation of mobile money transactions with the objective to generate a synthetic transactional data set that can be used for research into fraud detection. The data sets generated with PaySim can aid academia, financial organisations and governmental agencies to test their fraud detection methods or to compare the performance of different methods under similar conditions using a common public available and standard synthetic data set for the test.

We argue that PaySim is ready to be use as a tool to generate synthetic transactions that resemble the original and private data set supplied. By using PaySim we protect the privacy of the customers of the service at the same time that interesting results are possible to share with other researchers without the constrains and legal boundaries of the original data.

The results presented in the section 5. help to visually appreciate that the generated dataset captures the process and the frequencies of the different transaction types of the mobile money service.

Future work on the simulator is to add to the model fraudulent agents and run different scenarios to test the efficacy and accuracy of diverse fraud detection methods. We also want to make a synthetic data set available to other researchers and be able to compare and share diverse results.

ACKNOWLEDGMENTS

This work is part of the research project ”Scalable resource-efficient systems for big data analytics” funded by the Knowledge Foundation (grant: 20140032) in Sweden.

References

Chrystel Gaber, Baptiste Hemery, Mohammed Achemlal, Marc Pasquet, and Pascal Urien. Synthetic logs generator for fraud detection in mobile transfer services. In *2013 International Conference on Collaboration Technologies and Systems (CTS)*, pages 174–179. IEEE, may 2013. ISBN 978-1-4673-6404-1. doi: 10.1109/CTS.2013.6567225.

Volker Grimm, Uta Berger, Finn Bastiansen, Sigrunn Eliassen, Vincent Ginot, Jarl Giske, John Goss-Custard, Tamara Grand, Simone K. Heinz, Geir Huse, Andreas Huth, Jane U. Jepsen, Christian Jø rgensen, Wolf M. Mooij, Birgit Müller, Guy Pe’er, Cyril Piou, Steven F. Railsback, Andrew M. Robbins, Martha M. Robbins, Eva Rossmanith, Nadja Rüger, Espen Strand, Sami Souissi, Richard a. Stillman, Rune Vabø, Ute Visser, and Donald L. DeAngelis. A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, 198(1-2):115–126, September 2006. ISSN 03043800. doi: 10.1016/j.ecolmodel.2006.04.

023. URL http://linkinghub.elsevier.com/retrieve/pii/S0304380006002043.

Edgar Lopez-Rojas and Stefan Axelsson. Multi agent based simulation (mabs) of financial transactions for anti money laundering (aml). In Audun Josang and Bengt Carlsson, editors, *Nordic Conference on Secure IT Systems*, pages 25–32, Karlskrona, 2012a.

Edgar Alonso Lopez-Rojas and Stefan Axelsson. Money Laundering Detection using Synthetic Data. In Julien Karlsson, Lars ; Bidot, editor, *The 27th workshop of (SAIS)*, pages 33–40, Örebro, 2012b. Linköping University Electronic Press.

Edgar Alonso Lopez-Rojas and Stefan Axelsson. Social Simulation of Commercial and Financial Behaviour for Fraud Detection Research. In *Advances in Computational Social Science and Social Simulation*, Barcelona, 2014. ISBN 9789172952782.

Edgar Alonso Lopez-Rojas, Stefan Axelsson, and Dan Gorton. RetSim: A Shoe Store Agent-Based Simulation for Fraud Detection. In *The 25th European Modeling and Simulation Symposium*, number c, page 10, Athens, Greece, 2013.

S. Luke. MASON: A Multiagent Simulation Environment. *Simulation*, 81(7):517–527, July 2005. ISSN 0037-5497. doi: 10.1177/0037549705058073. URL http://sim.sagepub.com/cgi/doi/10.1177/0037549705058073.

S. F. Railsback, S. L. Lytinen, and S. K. Jackson. Agent-based Simulation Platforms: Review and Development Recommendations. *Simulation*, 82(9):609–623, September 2006. ISSN 0037-5497. doi: 10.1177/0037549706073695. URL http://sim.sagepub.com/cgi/doi/10.1177/0037549706073695.

Roland Rieke, Maria Zhdanova, Jurgen Repp, Romain Giot, and Chrystel Gaber. Fraud Detection in Mobile Payments Utilizing Process Behavior Analysis. In *2013 International Conference on Availability, Reliability and Security*, pages 662–669. IEEE, sep 2013. ISBN 978-0-7695-5008-4. doi: 10.1109/ARES.2013.87.

Balachandran Seetharam and Drew Johnson. Mobile Money's Impact on Tanzanian Agriculture. 2015.

AUTHORS BIOGRAPHY

MSc. Edgar A. Lopez-Rojas

Edgar Lopez is a PhD student in Computer Science at Blekinge Institue of Technology in Sweden and his research areas are Multi-Agent Based Simulation, Machine Learning techniques with applied Visualization for fraud detection and Anti Money Laundering (AML) in the domains of retail stores, payment systems and financial transactions. He obtained a Bachelors degree in Computer Science from EAFIT University in Colombia (2004). After that he worked for 5 more years at EAFIT University as a System Analysis and Developer and partially as a lecturer. He obtained a Masters degree in Computer Science from Linköping University in Sweden in 2011 and a licentiate degree in computer science (a degree halfway between a Master's degree and a PhD) in 2014.

MSc. Ahmad Elmir

Ahmad obtained a master in computer science with speciality in security from the Blekinge Institute of Technology. His master's thesis was about the design and construction of PaySim under the supervision of the main author of this paper. Previously he have studied natural sciences in the gymnasium for three years. His speciality was at computer science and programming. He has a keen interest for scientific inquiry in the domain of security as it is in his opinion an ever developing field. He have also worked for 9 months with software development in a corporation.

Dr. Stefan Axelsson

Stefan Axelsson is a senior lecturer at NTNU - Norwegian University of Science and Technology in Norway. He received his M.Sc in computer science and engineering in 1993, and his Ph.D. in computer science in 2005, both from Chalmers University of Technology, in Gothenburg, Sweden. His research interests revolve around computer security, especially the detection of anomalous behaviour in computer networks, financial transactions and ship/-cargo movements to name a few. He is also interested in how to combine the application of machine learning and information visualization to better aid the operator in understanding how the system classifies a certain behaviour as anomalous. Stefan has ten years of industry experience, most of it working with systems security issues at Ericsson.