ANALYSIS ON THE MODEL CHECKING AGENT FOR SMS MANAGEMENT SYSTEM VERIFICATION USING TEMPORAL LOGIC

Siti Dianah Abdul Bujang, Ali Selamat and Radziah Mohamad
Intelligent Software System Research Laboratory
Faculty of Computer Science & Information System
Universiti Teknologi Malaysia
81310 UTM Skudai,Johor, Malaysia.
anarel_03@yahoo.com, aselamat@utm.my and radziahm@utm.my

ABSTRACT

Nowadays, Short Message Service (SMS) is an emerging software mobile technology that support mobile-driven for business and telecommunication industry. The use of SMS has become an essential to most enterprises because the technology is known excellent. However, once SMS has been applied in distributed computing, the performance shows not given satisfied results. Extensive usage of SMS transaction has lead to the unseen faulty and errors. Also, the system is not able to support secure service to the users. This problem occurred because tremendous amount of SMS consume high bandwidth which could limit the network resources. Therefore, verification is needed to ensure the reliability of the system. This paper proposed the use of model checking agent approach for the verification of SMS Management System. The analysis of modeling shows the model checking agent is a basis approach to handle the SMS verification at the early stage of the software development.

Keywords: Short Message Service (SMS), Model Checking Agent, Verification.

1. INTRODUCTION

As an internet and mobile technology growth fast, it is one of potential technology to increase the market value for organizations. When we use the internet as a medium for managing commercial transactions, it can enhance the accessibility to a wide variety of information and services, and greatly facilitates remote payments. In fact, towards the evolution of the mobile technology, Short Message Service (SMS) is one of the notorious services that are capable to increase the performance of the conventional system. Thus, many companies are eager to convert the conventional system for mobile SMS application in order to have a better performance and service (Valentino, 2007), (Mee & Selamat, 2007) and (Shengguang, et al., 2008). Moreover, in the telecommunication industry especially, we always need a system that can reduce the time, cost and give high priority in security because they want the users to have satisfaction in services and performance. Therefore, the SMS become one of the technologies which support the requirement so far.

However, there are several cases of mobile systems have faced the unprecedented risk in operations especially involving on computing processes which resulting non trustworthiness for user point of view. They found out some problems occurred when many online ventures have been a victim to the operational processes because of poor security, poor data integrity control and inadequate management control once using the SMS. Others, Chaki et al. has claimed that now in information society, the correctness of computer software becomes one of the critical issues (Chaki, et al., 2008). The errors in complex software systems have caused large-scale economic losses in the past. Furthermore, Chaki et al. added that it is difficult to detect and fix the software bug in multi-threaded systems.

Therefore, for a distributed application such as SMS Management System, we assume that the system will encounter risk during a heavy workload of SMS transaction. Problems occur when tremendous amount of SMS need to share the network bandwidth which could limit the network resources. Then, the bandwidth and network traffic became congested (Timothy & Scott, 2000). Consequently, the system will be unstable and prone to errors. Therefore, verification is needed to ensure the reliability of the system. We propose the use of model checking agent approach for the verification of system that based on the integration of agent based system and model checking method. This approach is one of potential idea in order to ensure the SMS Management System can support robustness and verifiable to satisfy user.

In this research, we show the analysis of our model checking agent in a temporal logic approach. We have designed the SMS agent architecture and we test it using a model checker approach. The remainder of this paper is organized as follows: We present the related works in Section 2. Sections 3 showed the previous and propose architecture of the system. Section 4 and 5 respectively, showed the formal verification of SMS properties and experimental of result. Then, we conclude our paper in Section 6.

2. RELATED WORKS

2.1 SMS issues

Today's everyone used Short Message Service (SMS) to send and receive messages to retrieve the information in a fast way. It became one of popular communication technology that offers services to mobile devices including PDAs, all type of mobile phones, computers and any other terminals. It is an easy technology that been created in Europe by Global System for Mobile Communications (GSM) (http://www.etsi.org, 2004). Lately, there are billions of messages has been sent worldwide and the number is getting increase every year (Schusteritsch R., et al., 2005). However, when the SMS is applied in the application of business industry, we have anticipated several issues in the design challenges which are the limitation of the network bandwidth and network resources for communication between the mobile phones users and the system. Although, the SMS is a low cost consume money with high speed services, but somehow the undesirable congestion can probably effect the system management (http://cramsession.brainbuzz.com, 2004).

2.2 Formal Method and Model Checking

Nowadays, formal methods have an important role to represent the specification of software for complex system. It is rapidly becoming a promising automated method to enhance the accuracy and correctness of the software systems. The verification is based on the static analysis that able to analyze systematically and exhaustively (Eugenio D. S., et al., 2003). The known formal method model checking is a current evolving technology to evaluate the e-business process. Model checking provides an effective and efficient evaluation with counter-example to verify the correctness of complex software system (Bonnie B. A., et al., 2005). This method can effectively represent a systematically exhaustive mathematical models and infinite models to trace the errors. In verification using model checking, there are several model checkers can be used for testing our model. SPIN (Simple Promela Intepreter) can be written using Promela language and is representative of LTL model checker for communication protocol or concurrent software. SPIN provides safety and liveness properties such as deadlock, invalid end state and non-progress cycle (Bang Ki, et al., (2005).

Generally, to specify the property of a system in model checking, we use Computational Tree Logic (CTL) or Linear Temporal Logic (LTL). The logics specify the behaviors of a system based on time structure. In the case of our research, we use the LTL because the time sequence is assumes as linear. The model of the system is represent in Kripke structure as we proposed in (Selamat, A. & Bujang, A.S.D., 2008).

3. SYSTEM ARCHITECTURE

SMS Management System is one of e-business system that integrates with SMS for multi level marketing (MLM) services (Mee & Selamat, 2007). There are four main modules in this system; registration module, selling product, bonus calculation and group SMS for activities sharing. However, this architecture has been enhanced for greater performance by using our proposed architecture in Fig 1.

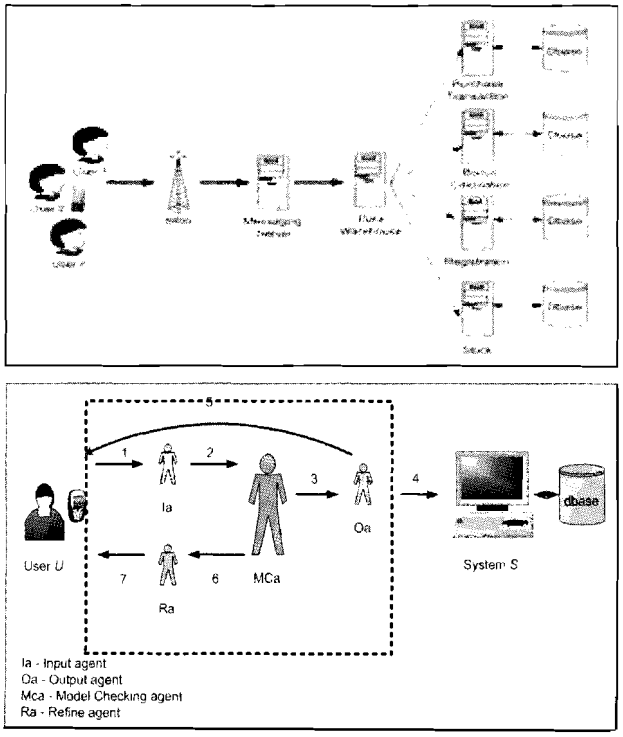

Fig. 1: The SMS Management System and the propose architecture

The propose architecture is the enhancement from the SMS Management System into agent- based system with the model checking agent as the manager to control the SMS system. We identified that the problems occurred in SMS Management System is when they are dealing thousands of user sending the messages to a system. The services

became slow and the probability of system to receive an invalid data is greater. As a result, the collision between the messages will increase the system errors. Therefore, the proposed of our model checking agent will be suggested to enhance the quality of services to check the correctness of the system design.

3.1 Modeling of Agent

In dealing with the problems in SMS Management System, we proposed a Model Checking Agent (MCa) with other supporting agents includes Input Agent (Ia), Output Agent (Oa) and Refine Agent (Ra). The MCa plays as a manager that will verify the incoming message. The message is verify based on three properties; format and syntax checking, byte transmission checking and timing checking. However, in this paper we only showed the verification of format and syntax checking as the experiment is still on going. The verification design of our MCa will then be tested using model checker SPIN.

Based on Fig. 1, we enhance the architecture into agent-based system with the role of all agents as the following below:

i. Input Agent (Ia)
Input Agent is a user interface in mobile phone has used as a medium for a users to interact with the SMS management system. The Ia will receive the request from user and assist the MCa to get the input from the user.

ii. Model Checking Agent (MCa)
The Model Checking Agent role as a manager of the whole system that will check and verify the SMS request whether does it satisfies the system requirement or could be otherwise. The requirement is verify based on the properties that we mentioned before. This process is done in order to ensure the system get a correct and valid message.

iii. Output Agent (Oa)
Output Agent will be reacted if the messages successfully verified from the MCa. Then the Oa will accept the messages and sent it to the system server. The status of the user request will automatically send in back propagation to the user for acknowledgment.

iv. Refine Agent (Ra)
However, if the messages are identified as failure, the refine agent or Ra will handle to do the refinement of any errors in the user request by sending 'failure status'. It means that the user has to send back their request again for new process. Otherwise, the message will be considered as violated and terminate automatically.

The flow chart of the agents' role can be showed as in Fig 2.

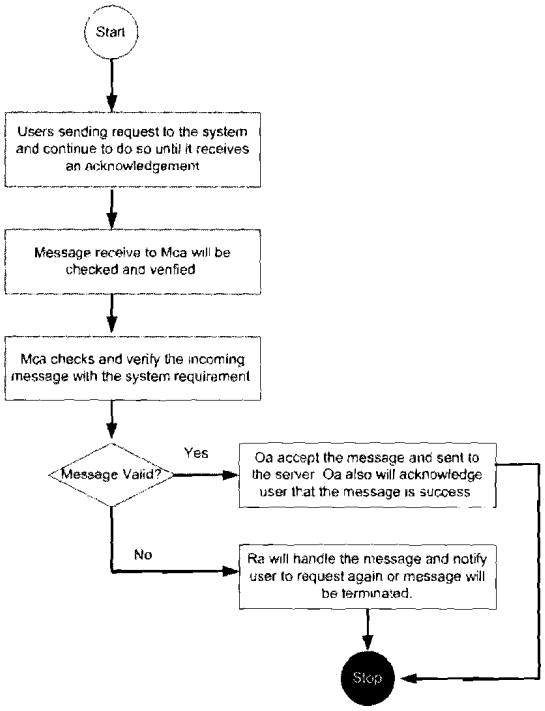

Fig. 2: The flow chart of the verification

4. FORMAL VERIFICATION OF SMS PROPERTIES

In model checking, we use linear temporal logic (LTL) to encode the formulae about the future path with modalities referring to time. It has two main types of properties that can be expressed using LTL which are safety and liveness. Safety properties is usually state that something bad never happens, while liveness properties is something good keeps happening. Therefore, we defined two properties in this paper: "no messages redundant" (safety) and "all messages send and receive eventually get through the system in valid format" (liveness).

Then, we verified the correctness of the SMS property based on format and syntax checking using SPIN. There are two paths for the issues of format and syntax checking which are:

1. Ia → MCa → Oa : Process checking and verifying is success.
2. Ia → MCa → Ra → Ia : Process checking and verifying is fail.

Based on the issues, MCa will checks and verify the format and syntax message in SMS Management System. MCa keeps track the message status as valid, invalid and fuzzy. It is denoted as v, i and f. Let say we represent the number of messages M in a situation of transmission. Then, a new transmission is made by the user as if $m \in M$ and $M \neq v$ or $m \in M$ and $M = i \cap f$, then the MCa wil response as "fail_message" and "refine_message". But, if $(m \in M = v = 1) \cap (i = f = 0)$, then the MCa will response as "success_message". The message then will be accepted by the system. The process is showed in the Fig. 3.

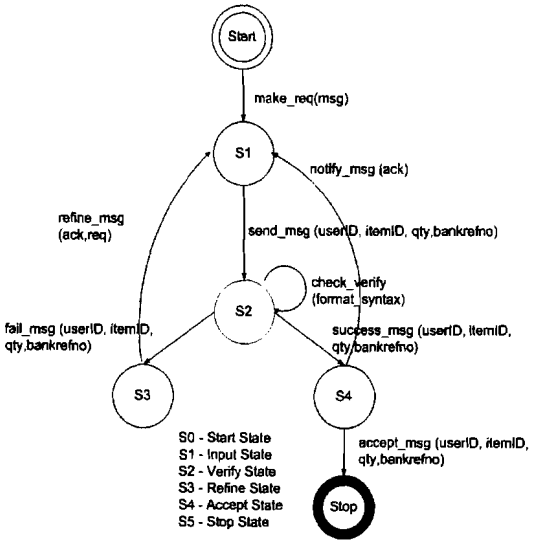

Fig. 3: Formal verification result of format and syntax checking

Based on Fig. 3 the property of the system must be specified based on the system requirement. The MCa has to follow a protocol to communicate using the formula of LTL. The formula used as follows:

$$1.\ G\left(\neg\left(\left(mk_{req}(msg)\right) \wedge \left(send_{msg}(uID, iID, qty, bnkref)\right)\right) \Rightarrow F\neg(accept)\right)$$

At the initial state, the user request will not be allowed and no transmission message is handled, then the path is not accepted.

$$2.\ G\left(\left(\neg\left(msg_{success}(uID, iID, qty, bnkref)\right)\right) U \left(refine_{msg}(ack, req)\right)\right)$$

The message will not be accepted as success until the refinement of message is done.

5. EXPERIMENTAL RESULT

The verification results in Fig. 4 shows that the both property satisfy the correctness of property as it is no assertion violations in the specification we done. However, we can find a significant difference between the safety and liveness properties with SPIN. The result determines that the format and syntax checking for the properties that we defined used 2.302 Mbyte of the computer memory. Although the result is identified as no assertion violation in the state of the system, but the SPIN has showed there is an error in some state for the refinement in the counter-example. We have to investigate and analyze the specification for the SPIN to find the reason why the result performs in the simulation. We assume that the source code and specification that we specified from natural language to PROMELA does not perform as well as the automata model that we designed. Therefore, the process of the verification is not complete.

```
Full statespace search for:
        never claim     - (none specified)
        assertion violations    +
        cycle checks    - (disabled by -DSAFETY)
        invalid end states      +
State-vector 36 byte, depth reached 5, ••• errors: 1 •••
2.302   memory usage (Mbyte)
```

Fig. 4: Formal verification result of format and syntax checking for safety and liveness properties

6. CONCLUSION

This paper explains a way to identify the problems which can cause a non trustworthiness of the system performance by using the model checking agent approach. It is important for the system to be designed accurately for system safety and not affect to the quality of service in telecommunication networks (QoS). However, in this paper, we have presented a mechanism of model checking agent specify and verify the SMS properties of syntax and format, the number of byte transmission and the time consume for each SMS transmission. We use Spin to show the correctness of the design with provided a counter-example that define sequences of processes that may lead to failure. However, during the number of states that can be verified is limited because due to the errors found in the SPIN. However, the improvement is still in the process of experiment as we attempt to make some adjustment of the specification. But the performance analysis using SPIN has given an effective testing tool to identify software bugs and errors at the early stage of the software development. Also, the proposed of model checking agent approach is able to be one of a new solution in order to overcome the verification issues especially in SMS Management System.

As in the future works, we will enhance the results with two properties that we defined for SMS quality checking.

ACKNOWLEDGEMENT

This work is supported by the Ministry of Science & Technology and Innovation (MOSTI), Malaysia and Research Management Center, Universiti Teknologi Malaysia (UTM) under the Vot 79200.

REFERENCES

Bang Ki S., Jin Hyun W., Yoo C. and Choi Jin Y., System Resource Utilization Analysis based on Model Checking Method, *An International Journal of Computing and Informatics* (Informatica), 2005.

Bonnie B. A., James. V. H., Paul B. L. and Scott L. S., Model Checking for E-Business Control and Assurance, *IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews*, vol. 35, No. 3, August 2005.

Chaki S., Clarke E., Sharygina N. and Sinha N., Verification of Evolving Software via Component, Substitutability Analysis, *Springer Science+Business Media*, LLC 2008.

Configuring SMS WAN Bandwidth http://cramsession.brainbuzz.com/articles/get-articles.asp?aid=805 , 2004.

Eugenio D. S., Francesco M. D., Marina M., Giacomo P., Web applications design and maintenance using Symbolic Model Checking, *Proceedings of the Seventh European Conference on Software Maintenance and Reeingineering* (CSMR'03) IEEE, 2003.

European Telecommunications Standards Institute, http://www.etsi.org, 2004.

Mee V. N. and Selamat A., SMS Management System for Direct Sales and Network Marketing. *Fifth International Conference on Information Technology in Asia* (CITA'07), 2007.

Selamat A. and Bujang A., The design of model checking agent for SMS Management System. *2nd KES In Symposium* (KES AMSTA), *Incheon, Korea, Springer-Verlag Berlin Heidelberg* 2008, LNAI 4953, pp. 813-821, 2008.

Shengguang M., Wei C., Gang L. Shitong W. and Liu W., An Asset management System based on RFID, WebGIS and SMS, *2nd International Conference on Ubiquitous Information Management & Communication* (ICUIMC'08), 2008.

Schusteritsch R., Rao S. and Rodden K., Mobile search with text messages:designing the user experience for google SMS. In *ACM Special Interest Group on Computer-Human Interaction (SIGCHI)*, (2005).

Timothy L., Scott A.D, Automatic Verification of Multiagent Conversations, *The Midwest Artificial Intelligent & Cognitive Science Conference*, (MAICS2000), 2000.

Valentino C.A, Progress Monitoring for the Field Operations of the 2007 Philipine Cencus of Population, *National Statistics Office*, Philipines, 10-12 December 2007.

Siti Dianah A. Bujang has received a B.Sc (Hons) in Computer Science from the Universiti Teknologi Malaysia and currently is pursuing her M.Sc. Computer Science by research under a supervision of Assoc. Prof. Dr Ali Selamat at the same university since 2007 until now. She has published papers in several international and local conference including the IEEE and Springer. Her research interests include formal method, software agent, software engineering process and application, software modeling and specification and e-business.

Ali Selamat has received a B.Sc. (Hons.) in IT from Teesside University, U.K. and M.Sc. in Distributed Multimedia Interactive Systems from Lancaster University, U.K. in 1997 and 1998, respectively. He has received a Ph.D. degree from Osaka Prefecture University, Japan in 2003. Currently, he is an associate professor and IT manager at the School of Postgraduate Studies, UTM. His research interests include software engineering, software agents, web engineering, information retrievals, genetic algorithms, neural networks and soft computing.

Radziah Mohamad has received a B. Sc. in Software Engineering from Sheffield Hallam University, UK. She pursued her study for M.Sc. in Computer Science and Ph.D. degree from Universiti Teknologi Malaysia (UTM). Currently, she is a lecturer at the Department of Software Engineering, Faculty of Computer Science & Information System (FSKSM), Universiti Teknologi Malaysia (UTM) from 2000-now. Her research interests include software agent, formal method, and software engineering and embedded real time system.