Formal Reliability analysis of Device Interoperability Middleware (DIM) based E-health system using PRISM

Usman Pervez*, Asiah Mahmood*, Osman Hasan*, Khalid Latif* and Amjad Gawanmeh†
*School of Electrical Engineering and Computer Science, National University of Sciences & Technology, Islamabad, Pakistan
Email: {usman.pervez,asiah.mahmood,osman.hasan,khalid.latif}@seecs.nust.edu.pk
† Department of Electrical and Computer Engineering, Khalifa University, UAE
Email: amjad.gawanmeh@kustar.ac.ae

Abstract—**Ensuring the correctness of middleware that ensures interoperability of various medical devices is one of the biggest challenges in the e-health domain. Traditionally, these Device Interoperability Middleware (DIM) are analyzed using software testing. However, given the inherent incompleteness of testing and the randomness of the user behaviours, the analysis results are not guaranteed to be accurate. Some of these inaccuracies in analysis results could even put human life at risk. In order to overcome these limitations, we propose to use a probabilistic model checker PRISM for analyzing DIM. The proposed approach allows us to rigorously verify reliability properties of the given DIM and thus allows the designers to make appropriate measures to design more reliable systems. For illustration, we formally analyze a middleware that uses the HL7 FHIR and ontology-based description of the devices and a communication protocol to bridge the gap in heterogeneity for dealing with different vendors and incompatible data formats.**

I. Introduction

One of the biggest challenges in the e-health domain is the interoperability of the underlying e-health systems due to the lack of standardization in the manufacturing of medical systems. The medical devices are diverse in nature and their functionality differs from device to device and the communication mechanism may also be different (e.g., Wifi or Serial port). To consider the impact of the interoperability problem, consider a hospital thats aims to setup its workflow using health information systems (HIS) [9]. This new setup can only be established if we are able to find medical devices that work on the same standards and support the same device integration mechanisms. Similarly, if a hospital aims to upgrade its existing e-health based medical system by adding a new medical device, such as a urine testing machine, then the new medical device must be compatible with the existing electronic environment and must follow the same standard that the other medical devices are following. In case of an unavailability of such a device, the hospital may have to upgrade all of its existing medical devices, which would certainly be a very undesirable solution for most hospitals.

One of the promising solutions to the above-mentioned problem is the development of a middleware that completely resolves the problem by bridging the gap between different standards of the medical devices. For example, if two devices using the serial port and WiFi are required to communicate with each other, then the communication can take place by introducing a middleware between them in such a way that it translates the output of the first device coming on the serial port to WiFI compatible data so that it can be received by the other device. Similarly, the middleware will also be responsible to translate the WiFi output data of the second device into serial data so that it can be received by the first device. This paradigm shift of overcoming the device interoperability problem from standardization of the workflows to the development of the Device Interoperability Middleware (DIM) [4] has shown a significant potential to solve the interoperability problem.

Considering the safety-critical nature of the medicine domain, ensuring the correct functionality of DIM is very important. In particular, if the middleware fails to translate the data from one communication standard to another standard, then this may lead to false results and hence false diagnostic reports of the patients will be produced which is extremely undesirable. Traditionally, the functionality of a middleware is checked by software testing. However, given the enormous number of possible scenarios in these DIM, they cannot be exhaustively tested due to computational power and memory constraints. Thus, the quality of DIM is judged based on a set of test vectors. This kind of incomplete testing of DIM can have serious consequences, including human deaths.

Formal methods [16], which use mathematical analysis methods in a computer, are capable of overcoming the inaccuracy limitations of simulation. Some prominent examples of formal methods based verification of healthcare systems include the verification of collaborative and agent based workflows in healthcare [8], [18], verification of electrocardiogram (ECG) biosensors in event-B [3] [2], FHIR standard based e-health system [27], software components in medical devices [5], [11], ambient assisted systems [7], or healthcare requirements [13]. The Communicating Sequential Processes (CSP) have been adopted as a formal method language to extensively formalize the system specifications and utilize it as a useful extension in the specification refinement in the system engineering lifecycle [12]. Similarly, a probabilistic model checker has been used to model and verify the treatment therapies of Tuberculosis and HIV [21]. Recent work shows some trials on using formal methods in the verification of software components in health care systems. For instance, model checking has been used to verify the reliability of software used in medical devices for an infusion pump [14]. Model-Based Testing has also been used to generate test cases for healthcare systems [28]. A formal model for e-Healthcare readiness assessment was also proposed in [26]. However, to the best of our knowledge, formal methods have never been used to analyze any DIM.

Given the safety-critical nature of the DIM, it is a dire need to assess its functionality, reliability and performance using formal methods. As a first step towards this direction, we propose to conduct the reliability analysis of DIM using probabilistic model checking. The usage of a probabilistic model checker allows us to capture the natural randomness found in the DIM models. The considered DIM has been developed as a middleware to integrate various medical devices that run on different communication mediums, including serial port, WiFi and bluetooth. This DIM has been installed in different hospitals of Pakistan and it enables automatic up-gradation of the medical systems by adding any new medical device that runs on either of the three communication mediums.

The main idea behind the proposed formal analysis is to develop a Markovian model of the medical system, including the corresponding DIM, and analyze it using the probabilistic model checker PRISM [23]. In particular, we use the Markov Decision Processes (MDP) [19] in PRISM to find the probability of occurrence of wrong results (failures) in the considered system having DIM installed. The proposed approach provides more accurate results than traditional counterparts due to the exhaustive exploration of a state-based model of the DIM based medical system.

The rest of paper is organized as follows: Section II describes some preliminaries about model checking and PRISM to facilitate the understanding of the paper. The considered health information system along with its reliability analysis is described in Section III. This is followed by the reliability analysis of two versions of the MDP based model of the considered health information system in Sections IV and V, respectively. Finally, Section VI concludes the paper.

II. Probabilistic Model Checking and PRISM

Probabilistic model checking is a variant of traditional model checking [6], where the probabilistic behavior of the given system is described using a Markovian model. This model can then be used to verify probabilistic properties. Many probabilistic model checkers, including PRISM [23], YMER [29], MRMC [20], VESTA [22] and ETMCC [15] are available. The main objective of the proposed work in this paper is to find failure and success probabilities, which are usually modeled as Markov Decision Process (MPD). Both, YMER and VESTA do not support steady state probabilities and thus cannot be used for our purpose [24]. The PRISM model checker has been reported as the most efficient one in terms of memory consumption compared to MRMC and ETMCC. It also supports a wider range of models, including Markov Decision Processes (MDPs), Discrete Time Markov Chains (DTMCs) and Continuous time Markov Chains (CTMCs) and thus has been selected for analyzing the DIM based HIS system [23].

The models in PRISM are described using a state-based language called the PRISM language. Modules and variables are basic components of the modeling language. A model can consist of a number of modules whose state at a given time is represented by the values of local variables defined in those modules. The values of local variables of all the modules define the overall state of the system. A set of guarded commands describe the behavior of each module. PRISM provides support for a variety of property specifications such as PCTL, CSL, LTL and PCTL*. For example, $S_{\geq 0.99}[\text{“}normal\text{”}$ is the steady state probability of *normal* state $\geq$ 0.99. PRISM supports verification and analysis of time based properties which we use for the time based analysis of Markovian models. These properties are analyzed by associating a certain reward with each state of the model through a reward structure.

III. Reliability analysis of a Typical Health Information System

The reliability of a typical Health Information System (HIS) is expected to increase when the DIM is used as a middleware to overcome Device Interoperability problem. To observe this increase in reliability, we first present a manual HIS system, as depicted in the Fig 1. which is typically found in the hospitals and evaluate its reliability by using the proposed model checking approach. This system provides the means of communication between various stake holders.

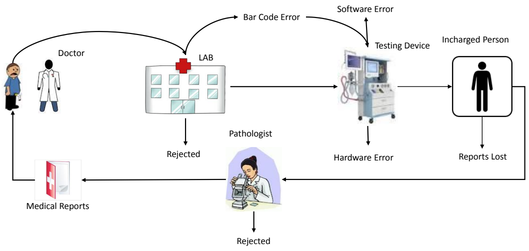

Fig. 1: A Typical Health Information System

As presented in the figure, when a patient visits the doctor, the doctor examines the patient and refers him to medical lab in order to undergo medical tests, such as blood test, urine test and glucose test. The lab collects the information of the patient, including his blood sample and generates his bar code. The blood sample is then fed in the medical device which performs the medical test. However, if the blood sample is found to be clotted or of low quantity, the patient's request is rejected. Upon successful completion of the medical test of the blood sample, the test reports are given to the person in charge of delivering them to a pathologist. During this process,

TABLE I: Transitional Probabilities

State Transitions	Probabilities	State Transitions	Probabilities
λ_0	1.0	λ_7	1-λ_4-λ_5=0.93
λ_1	0.02826	λ_8	0.4
λ_2	0.00174	λ_9	1-λ_8=0.6
λ_3	1-λ_1-λ_2=0.97	λ_{10}	0.25
λ_4	1.0	λ_{11}	0.75
λ_5	0.035	λ_{12}	1.0
λ_6	0.035		

the medical device may fail to perform the tests due to some hardware or software failures. Similarly, the reports may get lost while being delivered to the pathologist by the person in charge. Finally, the reports will be delivered to the patient after being successfully examined by pathologist.

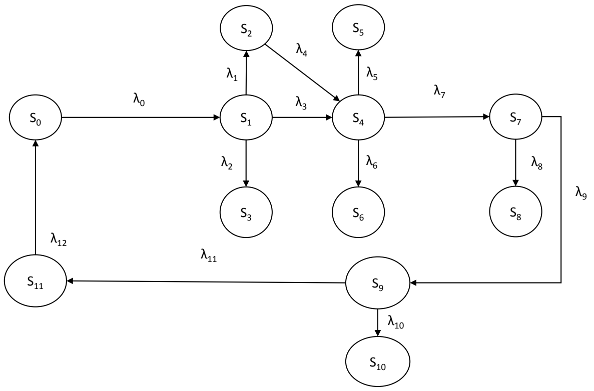

Fig. 2: State Machine of the manual HIS system

The behaviour of the underlying medical system is probabilistic due to the fact that the workflow transitions occur with some probabilities. In order to analyse the reliability of the overall medical system, which is the probability of successful delivery of the medical reports to the patient by the pathologist, the medical system is modeled with Markov Decisioning Process (MDP) [19]. MDP allows probabilistic decisions by including the appropriate state transition probabilities. The Markov Chain (MC) of the underlying health system is presented in Fig 2 and its transitional probabilities, which present the failure and successful probabilities of the transitional events, are depicted in the Table I. These transitional probabilities have been taken based on the statistics reported in [17] [10], and the probability of the reports being lost by the person in charge is considered to be 0.4.

In MDP, the transition from state ***S*** to a state ***S'*** is based on a probabilistic decision and depends on the present state ***S***. The transition probabilities for going from state ***S*** to state ***S'*** are governed by the following equation

$$P_a(S, S') = P_r(S_{t+1} = S' | S_t = S, a_t = a) \qquad (1)$$

where ***Pr*** is the transition probability and ***a*** is the action performed by the decision maker. The system is modeled in its mathematical form in terms of a Transition Probability Matrix ***P*** [1], which represents various transition rates. To calculate the probability of next state ***S'***, the Transition Probability Matrix ***P*** is multiplied with the current state probability as follows

$$P_r(S') = P_r(S) * P \qquad (2)$$

Based on our MDP model of the behavior of the HIS, described above, the following probabilistic properties, related to the system reliability, are verified:

$$Pmax = ?[F\ succ = 1] \qquad (3)$$

where ***Pmax*** is the output probability, ***F*** indicates eventually in the future, ***succ*** is a variable whose value is updated to 1 during the transition from state S_{11} to S_0. Similarly, we can find the probabilities of other successful as well as failure transitions by updating the value of the variable **succ** to 1, during those particular transitions.

By using the transition probabilities, as mentioned in Table I, we find the probability of success and failure of all the undergoing transitional events of the considered medical system and the results are presented in the Table II. These results show that the probability of the successful delivery of the medical reports to the patient by the pathologist is 0.41777.

TABLE II: Probability of Success and Failure

Lab Rejection	0.002	Human Error	0.637
Bar Code Error	0.047	Human Success	0.557
Device Hardware Error	0.060	Pathologist Rejection	0.239
Device Software Error	0.060	Pathologist Acceptance	0.417
Mchine Success	0.928	Successful Delivery	0.417

IV. Reliability analysis of DIM based Health Information System

After evaluating the reliability of a typical HIS system, we now move on to evaluate the reliability of a DIM based HIS [4] system with the aim to observe the increase in the reliability of the underlying HIS system. The workflow of a DIM based HIS system is presented in the Fig 3. This workflow is as the same as that of a typical HIS system with the difference that the responsibility of the person, who is in charge of delivering the medical reports to the pathologist, is now performed accurately by the automatic DIM middleware. The DIM in this example is composed of mainly two operations, which include a communication channel and a data mapping. The communication channel, which might be a serial port interface, a WiFi interface or a bluetooth interface, provides the medium to transfer the data taken from the medical machine output to the data mapper. The selection of the communication interface, i.e., serial port, WiFI or bluetooth, depends on the communication standard of the medical machine installed. If the machine has been designed to communicate through a serial port, the serial port interface will be used as a communication channel to communicate the data. Similarly, the wifi interface and the bluetooth interface will be used for the wifi and bluetooth compatible devices, respectively. This automatic selection of the communication interface has significantly resolved the device interoperability problem and thus facilitates new setups as well as easy upgradation of the HIS systems. For example, if the HIS system of a hospital only posses a blood testing medical device, which communicates only by a serial port, then this HIS system can be easily upgraded by adding any new medical device, such as a urine testing medical device, without taking care of the communication standard (i.e., serial port, WiFI, or bluetooth) being followed by the new device. The device mapper finally maps the raw data, as received from the communication channel, to the HL7 standard based diagnostic reports. It has the capability to understand the received raw data regardless of the format of the data i.e., serial port data format, WiFi data format or bluetooth data format. The diagnostic reports are then delivered to the pathologist automatically. The underlying HIS system is very efficient in terms of automatic reports delivery but it does not guarantee accuracy and perfection. Failures in the system, such as communication failure and data mapping failure, may result into a fatal loss. To avoid these failures, the system must be pre-tested before installation. We propose a reliability evaluation mechanism of the considered system by utilizing MDPs in the proposed methodology. We have developed the MC of the system, presented in the Fig 4, and its transitional probabilities are depicted in Table 3. These transitional probabilities have been taken based on the statistics reported in [17] [10], whereas the probability of the communication channel failure has been considered to be 0.1.

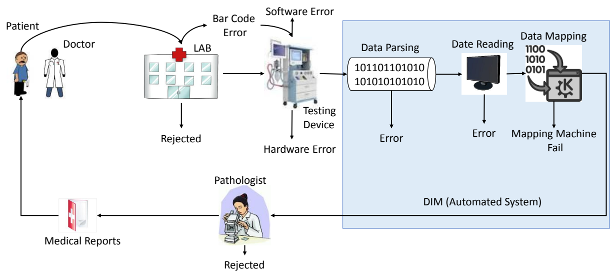

Fig. 3: DIM based HIS

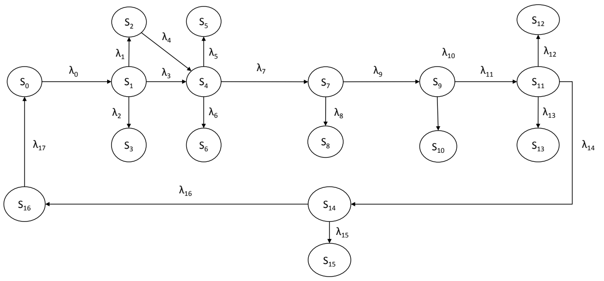

Fig. 4: State Machine of DIM based HIS system

We verified the reliability of the DIM based HIS system in terms of successful delivery of the diagnostic report to the patient using PRISM as follows:

$$Pmax = ?[F\ succ = 1] \tag{4}$$

TABLE III: Transition Probabilities

State Transitions	Probabilities	State Transitions	Probabilities
λ_0	1.0	λ_9	1-λ_7=0.9
λ_1	0.02826	λ_{10}	0.2
λ_2	0.00174	λ_{11}	1-λ_6=0.9
λ_3	1-λ_1-λ_2=0.97	λ_{12}	0.065
λ_4	1.0	λ_{13}	0.065
λ_5	0.035	λ_{14}	1-λ_{11}-λ_{12}=0.87
λ_6	0.035	λ_{15}	0.25
λ_7	1-λ_4-λ_5=0.93	λ_{16}	1-λ_{14}=0.75
λ_8	0.1	λ_{17}	1.0

TABLE IV: Probability of Success and Failure

Lab Rejection	0.00308	Data Reading Failure	0.29637
Bar Code Error	0.04904	Data Reading Success	0.66843
Device Hardware Error	0.06196	Mapping Machine Error	0.07705
Device Software Error	0.06196	Mapping Algorithm Failure	0.07705
Mchine Success	0.92838	Mapping Success	0.58153
DIM Success	0.58153	Pathologist Rejection	0.25784
Communication Failure	0.16465	Pathologist Acceptance	0.43615
Communication Success	0.83554	Successful Delivery	

The value of the variable ***succ*** is updated to 1 during the transition form state S_{16} to state S_0. Similarly, we can find the probabilities of other successful as well as failure transitions by updating the value of the variable **succ** to 1, during those particular transitions. Table IV presents the probability of successes as well as failures of all the events in the DIM based HIS. The results indicate that the reliability of the DIM based HIS system, which is the probability of successful delivery of the diagnostic reports to the patient, is 0.43615. It has been observed that the reliability of the DIM based HIS system is higher that the reliability of the typical HIS system, given in the previous section.

V. Reliability analysis of TMR enabled DIM based Health Information System

Based on the obtained results, as presented in the previous section, the DIM middleware has resulted in increasing the overall reliability of the system, but it has been noticed that this increase in reliability is not significant. Since, the HIS systems are very sensitive and require high accuracy due to the fact that its performance and reliability has a direct effect on the lives of the patients, there is a dire need to increase this reliability up to some acceptable level. For the purpose, we propose some modifications in the DIM system by leveraging upon the strengths of the Triple Modular Redundancy (TMR) mechanism [25].

TMR is a fault tolerating mechanism that is used frequently in safety-critical systems where a single fault in the system may lead to some drastic situations. This mechanism has the capability to tolerate a single fault in the system and thus the reliability of the underlying system increases. A typical TMR mechanism has been presented in Fig 5. In TMR, the critical process is performed separately by three identical resources functioning in parallel, whereas the output of all the three resource are fed into a voter. The voter checks these outputs and decides the final output on the basis of a majority voting system. If all the three resources produce the same output, the voter will consider the system flawless and produce the same output, as produced by the resources. However, if any one of the resource produces a different output, due to some unknown fault, in comparison to the remaining two resources, the voter will consider this resource faulty. It will tolerate this fault by masking it and keep the system functioning by producing the output, as produced by the remaining two resources. For the case, when all the resources produce different outputs, the voter will consider the whole system faulty. The MC of a common TMR mechanism is presented in Fig 6. In the figure, state 1 shows that that all the three resources are functioning properly. State 2 shows that only one system is faulty, however the system b is still functioning. If any other resource fails from this point, the whole system is go into state ***F***. In the figure, the term λ represents the failure rates.

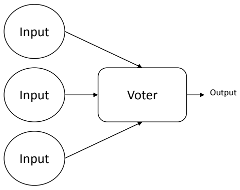

Fig. 5: Triple Modular Redundancy

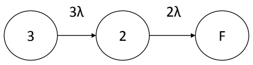

Fig. 6: MC of a TMR mechanism

As discussed in the previous section, the communication channel of the DIM middleware can communicate through any of the three communication interfaces i.e., serial port, WiFi and bluetooth. Moreover, only one communication interface is used at a time to communicate the data from medical device output to the data mapping. With the aim to increase the reliability of communication channel in terms of successful communication, we propose to use all the three communication channels in parallel, while using the concepts of TMR mechanism. This proposed modification of the DIM middleware can significantly increases the reliability of the underlying HIS system by enhancing the reliability of the communication channel. This choice would obviously need devices that can communicate via all three communication mediums and is thus the cost of the additional reliability gained. The Markov chain of the modified DIM based HIS system is presented in the Fig 7 and its transitional probabilities are presented in the Table V, whereas the failure probabilities of all the communication interfaces are considered to be 0.1.

TABLE V: State Transitional Probabilities

State Transitions	Probabilities	State Transitions	Probabilities
λ_0	1.0	λ_{11}	1-2λ=0.8
λ_1	0.02826	λ_{12}	1.0
λ_2	0.00174	λ_{13}	0.2
λ_3	1-λ_1-λ_2=0.97	λ_{14}	1-λ_{11}=0.8
λ_4	1.0	λ_{15}	0.065
λ_5	0.035	λ_{16}	0.065
λ_6	0.035	λ_{17}	1-λ_{13}-λ_{14}=0.87
λ_7	1-λ_5-λ_6=0.93	λ_{18}	0.25
λ_8	3λ=0.3	λ_{19}	1-λ_{14}=0.75
λ_9	1-λ_4-λ_5-λ_6=0.7	λ_{20}	1.0
λ_{10}	2λ=0.2		

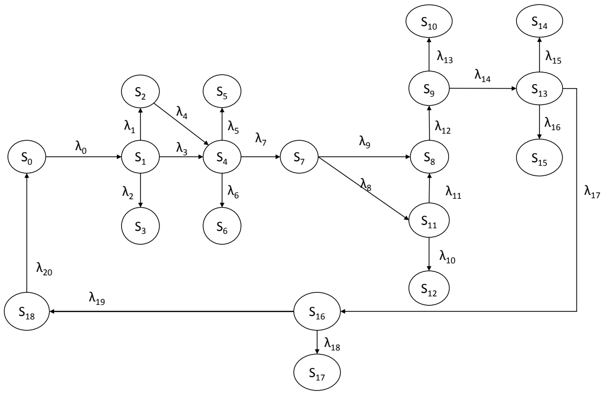

Fig. 7: State Machine of Modified DIM based HIS system

In Fig 7, the state S_4 represents the status of the medical device machine. If the tests are successful, the state machine moves to the state S_7, which indicates that all the three communication channels are functioning successfully. If one of the communication channel fails, the state machine will enter the state S_{11}. Finally, if more than one communication channel fail, the state machine will move to the fail state, which is represented by S_{12}. By using the transitional probabilities, presented in the Table V, we obtained the reliability results which are depicted in the Table VI. Results indicate that the overall reliability of the modified DIM based HIS system has been increased to 0.4555.

To the best of our knowledge, the verification of properties, like the probabilities of success and failures, reported in this paper, have not been verified in the context of any DIM before. Moreover, traditional techniques, like numerical methods and simulation, cannot match the precision of results obtained by the proposed approach for systems with such a large number of state-transitions.

TABLE VI: Probability of Success and Failure

Lab Rejection	0.00319	Data Reading Failure	0.32056
Bar Code Error	0.0507	Data Reading Success	0.6981
Device Hardware Error	0.06417	Mapping Machine Error	0.08334
Device Software Error	0.06417	Mapping Algorithm Failure	0.08334
Machine Success	0.92838	Mapping Success	0.60738
DIM Success	0.60738	Pathologist Rejection	0.27889
Channel Failure	0.4215	Pathologist Acceptance	0.4555
Communication Failure	0.1023	Successful Delivery	0.4555
Communication Success	0.87263		

VI. Conclusion

This paper presented a formal reliability analysis for a typical DIM based HIS system in a hospital setting. The main contributions of the paper include the development of the Markovian models for the typical HIS system, DIM based HIS system and TMR enabled DIM middleware. These behaviors are then used to develop the MDP models, using the PRISM language, and check various probabilistic properties of our system. The paper also identifies some key reliability assessment properties of the DIM based HIS system that can be formally verified by PRISM. The proposed approach has been found to be more accurate and scalable and it also allows us to verify many novel reliability aspects compared to other existing reliability analysis approaches for health care standards. We plan to use the proposed approach to analyze the reliability of other standards of Device Interoperability problem as well. We also plan to see the impact of having **n** level-redundancy in the models on the overall system reliability.

References

[1] O.O. Aalen and S. Johansen. An Empirical Transition Matrix for Non-Homogeneous Markov Chains based on Censored Observations. *Scandinavian Journal of Statistics*, 5(3):141–150, 1978.

[2] H. Al Hamadi, A. Gawanmeh, and M. Al-Qutayri. A Verification Methodology for a Wireless Body Sensor Network Functionality. In *Biomedical and Health Informatics (BHI)*, pages 635–639, 2014.

[3] H. Al-Hamadi, A. Gawanmeh, and M. Al-Qutayri. Formalizing electrocardiogram (ecg) signal behavior in event-b. In *e-Health Networking, Applications and Services (Healthcom)*, pages 55–60. IEEE, 2014.

[4] A.Mahmood, F.Ahmed, K.Latif, H.Mukhtar, and A.Raza. Middleware for Medical Device Interoperability using Ontology-based Description and Mapping. *Technical Report, National University of Sciences and Technology, Pakistan*, 2015. http://semr.seecs.nust.edu.pk/downloads/middleware.pdf.

[5] S. M. Babamir and M. Borhani. Formal Verification of Medical Monitoring Software Using Z Language: A Representative Sample. *Journal of Medical Systems*, 36(4):2633–2648, 2012.

[6] C. Baier and J. Katoen. *Principles of Model Checking*. MIT Press, 2008.

[7] K. Benghazi, M. V. Hurtado, M. L. Rodrguez, and M. Noguera. Applying Formal Verification Techniques to Ambient Assisted Living Systems. volume 5872 of *LNCS*, pages 381–390. Springer, 2009.

[8] C. Bertolini, Z. Liu, M. Schaf, and V.Stolz. Towards a Formal Integrated Model of Collaborative Healthcare Workflows. In *Foundations of Health Informatics Engineering and Systems*, volume 7151 of *LNCS*, pages 57–74. Springer, 2012.

[9] S. Brown. Networked health information system for monitoring food intake, October 7 2003. US Patent App. 10/605,548.

[10] J.C. Dale and D.A. Novis. Outpatient phlebotomy success and reasons for specimen rejection. *Archives of pathology & laboratory medicine*, 126(4):416–9, 2002.

[11] Z. Daw, R. Cleaveland, and M. Vetter. Formal Verification of Software-Based Medical Devices considering Medical Guide-lines. *Computer Assisted Radiology and Surgery*, 9(1):145–53, 2014.

[12] O. Faust, U. R. Acharya, and T. Tamura. Formal Design Methods for Reliable Computer-Aided Diagnosis: A Review. *IEEE Revisions in Biomedical Engineering*, 5:15–28, 2012.

[13] A. Gawanmeh. An Axiomatic Model for Formal Specification Requirements of Ubiquitous Healthcare Systems. In *Consumer Communications and Networking*, pages 898–902. IEEE, 2013.

[14] J. F. Groote, A. Osaiweran, and J. H. Wesselius. Analyzing the Effects of Formal Methods on the Development of Industrial Control Software. In *Software Maintenance*, 2011.

[15] J. Meyer-Kayser H. Hermanns, J. Katoen and M. Siegle. ETMCC: Model Checking performability properties of Markov Chains. In *Dependable Systems and Networks*, pages 673–673, 2003.

[16] O. Hasan and S. Tahar. Formal verification methods. *Encyclopedia of Information Science and Technology, IGI Global*, pages 7162–7170, 2015.

[17] R. Hawkins. Managing the pre-and post-analytical phases of the total testing process. *Annals of laboratory medicine*, 32(1):5–16, 2012.

[18] M. Hoogendoorn, M. C. Klein, Z. A. Memon, and J. Treur. Formal Verification of an Agent-Based Support System for Medicine Intake. 25:453–466, 2009.

[19] C.C. White III and D.J. White. Markov Decision processes. *European Journal of Operational Research*, 39(1):1–16, 1989.

[20] M. Khattri J. Katoen and I.S. Zapreev. A Markov reward Model Checker. In *Quantitative Evaluation of Systems*, pages 243–244, 2005.

[21] R. Jetley, S. PurushothamanIyer, and P. L. Jones. A Formal Methods Approach to Medical Device Review. *IEEE Computer Journal*, 39(4):61–67, 2006.

[22] M. Viswanathan K. Sen and G. Agha. Vesta: A statistical Model-Checker and analyzer for probabilistic systems. In *Quantitative Evaluation of Systems*, pages 251–252, 2005.

[23] M. Kwiatkowska, G. Norman, and D. Parker. PRISM 4.0: Verification of Probabilistic Real-time Systems. In *Computer Aided Verification*, volume 6806 of *LNCS*, pages 585–591. Springer, 2011.

[24] Axel Legay, Benot Delahaye, and Saddek Bensalem. Statistical model checking: An overview. In Howard Barringer, Ylies Falcone, Bernd Finkbeiner, Klaus Havelund, Insup Lee, Gordon Pace, Grigore Rou, Oleg Sokolsky, and Nikolai Tillmann, editors, *Runtime Verification*, volume 6418 of *Lecture Notes in Computer Science*, pages 122–135. Springer Berlin Heidelberg, 2010.

[25] R.E. Lyons and W. Vanderkulk. The use of triple-modular redundancy to improve computer reliability. *IBM Journal of Research and Development*, 6(2):200–209, 1962.

[26] S.O. Oio, O.O. Olugbara, G. Ditsa, M.O. Adigun, and S.S. Xulu. Formal Model for e-Healthcare Readiness Assessment in Developing Country Context. In *Innovations in Information Technology*, pages 41–45, 2007.

[27] U. Pervez, O. Hasan, K. Latif, S. Tahar, A. Gawanmeh, and M.S. Hamdi. Formal reliability analysis of a typical fhir standard based e-health system using prism. In *e-Health Networking, Applications and Services, 2014 IEEE*, pages 43–48. IEEE, 2014.

[28] M. Vieira, X. Song, G. Matos, S. Storck, R. Tanikella, and Bill B. Hasling. Applying Model-Based Testing to Healthcare Products: Preliminary Experiences. In *Software Engineering*, pages 669–672. ACM, 2008.

[29] H. Younes. Ymer: A Statistical Model Checker. In *Computer Aided Verification*, volume 3576 of *LNCS*, pages 429–433. Springer, 2005.