Verification and Validation of Cyber Physical System using "Probabilistic Automata"

Pritish Samant[1]

Contents

[1] pritish-sanjay.samant@stud.hshl.de

Abstract:

Cyber Physical systems(CPS) is combination of Physical Systems and computer network. Behaviour of a Cyber Physical system defines its usability and growth. Timing properties are very important to understand the behaviour of a system. The practical use of a system depends on its verification and validation. There are various approaches to verify and validate the model of a system. Here we will be using Probabilistic automata to verify and validate the model or a system.

1 Motivation

Cyber Physical system(CPS) is one of the most important technology which can shape the future. Combination of different processes like software and hardware constitutes Cyber Physical System(CPS). There are many fields where we can use such system's intelligence and try to decrease the human error factor. CPS is a complex architecture. To build its hardware error free as well as compatible with respective software for integration is extremely difficult task. Even if a cyber physical system is built, its maintenance and energy requirements are enormous. Even though with such difficulty, such systems are built more often. There are researchers going on with respect to scalability, usability and many more aspects of a cyber physical system(CPS).

The characteristics of CPSs are first presented, and the research developments are summarized from several aspects, including energy control, secure control, transmission and management, control technique, system resource allocation, and model-based software design[Sh11].We have followed a model-based strategy for the building of reliable and ideal CPS for more than 20 years, accompanied by tools for analyzing models with sound semantic foundations that are based on effective algorithms and data structures.[La17]. In order to analyse the model of a cyber physical system(CPS), we would need some kind of tool or mathematics to deploy system correctly. According to [MSW16],"The CPS must be able to overcome the system uncertainty, scalable and tolerant to threat". For this reason, the system must be checked and tried and tested beforehand.

The different approaches to check the Cyber physical system(CPS) are nothing but the models created using mathematical equations and timing behaviour of the system. If the CPS fits the approach then the CPS can be put in practical use. Here, we will use Probabilistic Automata approach to test the model of Cyber physical System to test its efficiency. In simple words, the stochastic mathematics and timing automation will determine the efficiency of a model.This is an important step in deciding the usability, scalability as well as accuracy of a Cyber Physical System(CPS). In this paper, we will be testing a model of a CPS using a formal approach with the help of an use case just to understand the working of the formal method or approach which is used to test a Cyber Physical System(CPS).

2 Foundation

2.1 Verification and Validation of CPS

Cyber Physical System(CPS) is a complex structure and to test its productivity, usability and efficiency in the physical world is an important task. Validation of the system is to establish that a formal method exists to check the system. Whereas verification of the system is to prove the system's accuracy using the formal method. Debugging CPS is important, but the intertwining of the cyber and physical worlds makes it very difficult. Regarding CPS verification and validation, there are a number of widely held notions and misunderstandings, but relatively little research has been done to comprehensively elucidate these riddles [ZJ15].Because of this, research into CPS verification and validation primarily operates in the shadows, focusing on problems that the CPS development community may or may not find important [ZJ15].

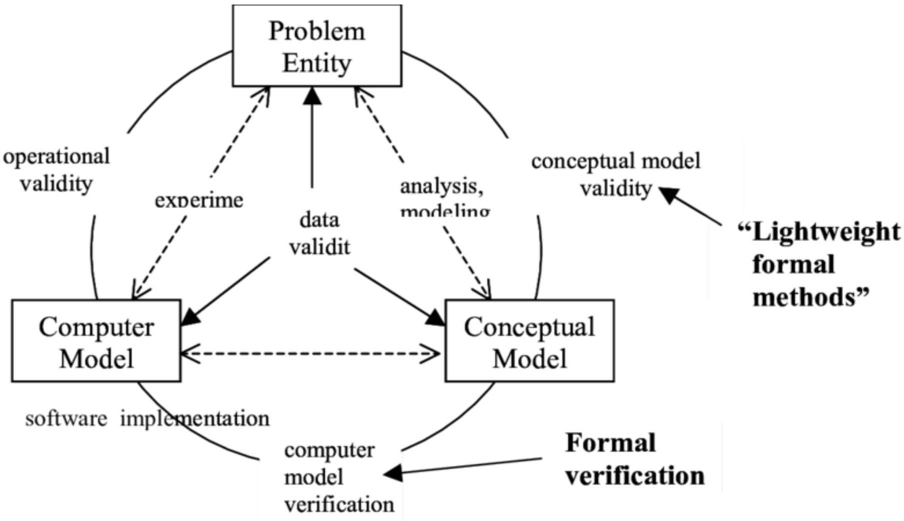

Fig. 1: Verification and Validation of CPS[KC02]

In the above Fig.1, we can see the process of validation and verification using formal method. Here, the problem entity is the system to be validated and tested. For which a conceptual model is prepared using light formal methods. This is also called as conceptual model validity. This is pat of validation. Then he conceptual model is turned into computer model using software implementation also called as formal verification. This is called as computer model verification and this is part of Verification. Further, this is experimented and the system is tested. This is called as operational validity. This system can again be tested using the same above procedure. This cycle represents the validation and verification of a Cyber Physical System(CPS). Further we will discuss one of the formal methods used to test the system called Probabilistic Automata.

2.2 Probabilistic Automata

According to [St02],”A Probabilistic Automata consists of four components:
1. A set S_A of states,
2. A nonempty set $S_A^0 \subseteq S_A$ of start states,
3. An action signature $sig_A = (V_A . I_A)$, consisting of external and internal actions respectively.
We require $V_A\ and\ I_A$ to be disjoint and define the set of actions as $A\ ct_A = V_A \cup I_A$.
4. A transition relation $\triangle_A \subseteq S_A\ x\ Act_A\ x\ Distr(S_A)$.
Again, we write $s \rightarrow A\ \mu\ for(s, a, \mu) \in \triangle_A$. Furthermore, we simply write $s \rightarrow As'\ for\ s \rightarrow As' \longmapsto 1$ ”
Probabilistic Automata is a model checking technique. According to [De11],By Definition, ”A probabilistic automaton (PA) is a tuple (S,A,L,AP,V,s0), where S is a finite set of states with the initial state s0S,A is a finite set of actions, L: S×A×Dist(S)→B2 is a (two-valued transition) function, AP is a finite set of atomic propositions and V:S→2AP is a state-labeling function".

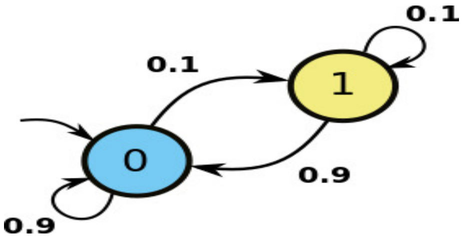

Fig. 2: Example of Probabilistic Automata [dFC20]

In the above Fig.2, it depicts the probabilistic automata. It is feasible to create probabilistic automata by permitting the transition probabilities to take values lower than 1, which enables the hypothetical agent to move randomly while obtaining a variety of patterns [dFC20].Here are two states 0 and 1. In this automata, the node 0 randomly goes to each node for infinite times with pattern 0 occurring more times than pattern 1. This is a basic example of Probabilistic Automata. Using this logic, we can check the model of a system.

3 Use Case

3.1 Iot enabled Smart washing machine

There can be several use cases for verification and validation of Cyber Physical Systems using Probabilistic automata. Here, we will see one of such use case where the system can be verified and validated using probabilistic automata. We will see an example of IoT enabled washing machine. According to [Ve21], ”In our scenario, we took IoT enabled washing machine consisting of nine states OFF, ON, LOADTYPE, COTTON, SYNTHETIC, MIXED, SPIN, RINSE, DRY and five input signals On, Off, C, S, M, W where On and Off for device turned on and tuned off respectively, C to Cotton-type clothing, S to Synthetic-type clothing, M to Mixed-type clothing and W to Wash. ON and OFF indicates if the washing machine is on or off. If the machine receives any other input signal in the “OFF” state it will result in anomaly. Now in the state machine “ON” has two choices, if the clock value X is greater than 30 s and less than 60 s (30<=X<=60) with probability P=0.9, the first alternative is to “LOADTYPE”. The second choice, if the clock value (30<=X<=60) with likelihood P=0.1, is to switch to ÖFF"mode. The probabilities of getting cotton, synthetic, and mixed-type clothing are 0.3, 0.2, and 0.5".

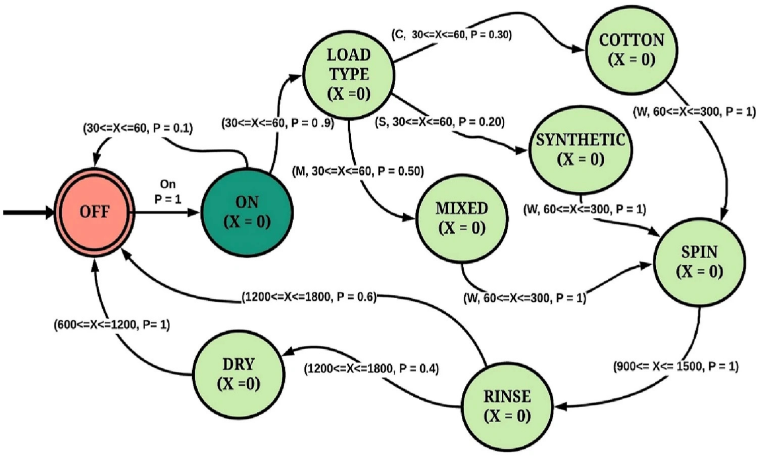

Fig. 3: Iot enabled Smart washing machine [Ve21]

If the washing machine received signal for between 30<=X<=60, then the event is considered as a legal event. other wise the event is an anomaly. between the time frame of 30s and 60s, the machine can receive inputs for cotton, synthetic and mixed from the user. After that, the machine goes to spin with probability P=1. Depending on the type of spin and the weight of clothes in the machine, the machine spins in the time frame of 900 s<=X<=1500 s. After this, the machine moves to rinse state. From here, there are two possibilites that the machine can go to dry state of off state depending on the user input. Since there is no other option available in "DRY"mode, the probability (P) is 1, and the state machine switches from ÖFF"to "DRY"when the clock (X) value is higher than 600 s and less than 1200 s [Ve21].

3.2 Vehicular ad-hoc network(VANET)

3.2.1 Modeling in Vanet

The vanet is a system or tool made for vehicles and these vehicles are connected wirelessly in a network to communicate with each other. In this paper we will see the modeling for vanet. Vehicles communicate with each other in a network. A vehicle also needs to check the surroundings and if there are any obstacles, it need to transmit an emergency message to rest of the vehicles. we will focus on probability this emergency transmission messages between vehicles. Example of this is given in Fig.4.

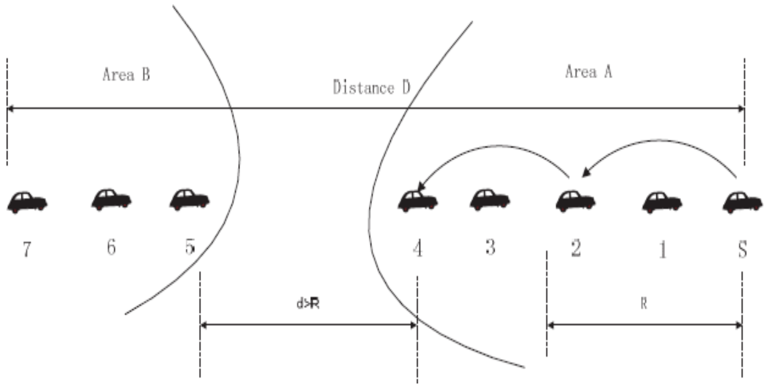

Fig. 4: Information Exchange in Vanet [LWL14]

s

Suppose a vehicle is moving from A to B and if there is an emergency message to transmit, it will transmit from B to A. The source of the emergency message is given as Nodes S and the area between node 4 and Sn is one network and the area between node 5 and node 7 is a different network. There should be a technique to ensure that the message can be delivered safely because the minimum distance between the divided areas (between nodes 4 and 5) is d>R [LWL14]. Now we will model the two parts of the system. The model will be of sending node and receiving node. The sending node S will send the message and it is transmitted in steps. If the transmission is timed out the the message is discarded and till the time out the message can be transmitted to itself if the the receiving is no able to receive till the time success message is transmitted from the node to the sending node S.

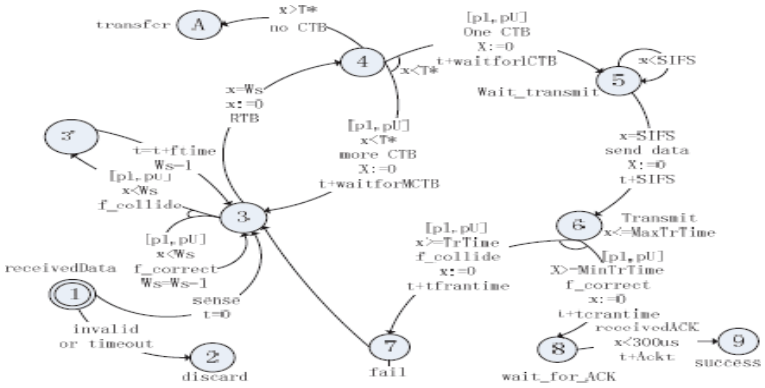

Fig. 5: Sending node [LWL14]

s

According to [LWL14], "we use interval probability [pL,pU] to express the transformation from node 4 to node 5. The definitions of pLand, pU are as follows:

pL=min{(n$_{min}/(cw^{*}N_s))^{*}pow(2.718,(-n_{min}/cw^{*}N_s)),$
$(n_{max}/(cw^{*}Ns))^{ast}pow(2.718,(-n_{max}/(cw^{*}Ns))),1\}$
$pU = min\{max(n_{min}/(cw^{*}N_s))^{*}pow(2.718,(-n_{min}/cw^{*}N_s)),$
$(n_{max}/(cw^{*}Ns))^{ast}pow(2.718,(-n_{max}/(cw^{*}Ns))),1\}$”

s

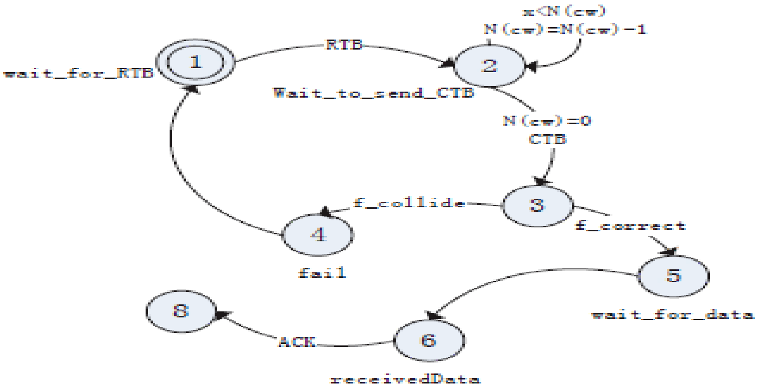

Fig. 6: Receiving node [LWL14]

s

According to [LWL14], "For the transmission from node 4 to node A, we use the interval probability to represent the uncertainty. The code in Prism is as follows:

[receiveNoCTB] (sl=4)& (xl=T) ->(sl$' = A)\&(xl' = 0)$”

3.2.2 Verification for usecase in Vanet

Formal verification is done to check if the model works correctly or not. In other words, if we have a cyber physical system and then we build a model that can prove the system and also increase its efficiency. This is a standardize model for various types of the same system than we have to check if the model performs correctly or not. This is the part where we check the model either through stochastic modeling or by using tools. This is called formal verification of a model. Concerning this paper, in vanet we see emergency messages transmitted and for this we created a model for vanet which is divided into two parts, model for sending node and receiving node. For this, we are using prism tool for model verification.

According to [LWL14], "We use PTCTL to describe the property. The syntax of PTCTL is as follows:

$(\beta; :=$ a$|\xi|_\eta \phi|\phi \vee \phi|$z.$\phi|P_{\sim\lambda}[(\beta]$”

And probability of Success is:

P=1-(1-P$_0) \times (1 - P_1) \times (1 - P_2) \ldots (1 - P_\mathrm{n})$[LWL14].

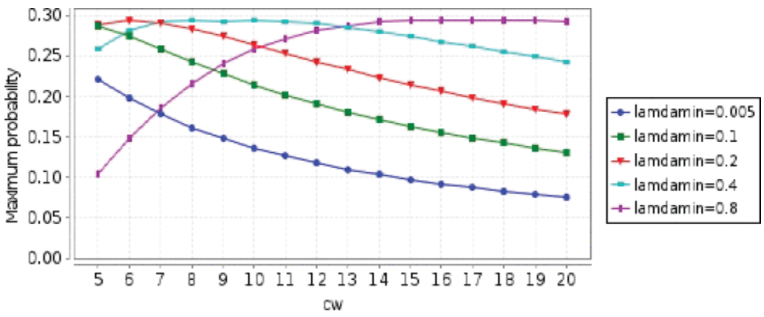

Fig. 7: RProbability of success [LWL14]

s

For this paper, the parameters used are transmission radius R, Ns, lambda function, v1 and the competition window cw[LWL14]. This formal verification code can be run against various test cases which are either statistical or behavioral aspect and also check its reliability, correctness and usability. Using this statistical inputs probability of success that a message is transmitted is done using formal verification method with the help of prism tool. It is also evident from the above example that large node densities can result in low success which increases the time in transmission of message [LWL14].

4 Conclusion

In the above use case, we used the probabilistic automata for anomaly detection of an IoT enabled washing machine and Vanet. We have also seen the verification method using prism tool. Similarly we can use the same method to test various other systems to test its ability and efficiency. As we can see, it is very important to test a system after its modelled and this can be done using various other techniques and probabilistic automata is one of them and a very important model checking approach to test systems.

5 Declaration of Originality

I, ..., herewith declare that I have composed the present paper and work by myself and without the use of any other than the cited sources and aids. Sentences or parts of sentences quoted literally are marked as such; other references with regard to the statement and scope are indicated by full details of the publications concerned. The paper and work in the same or similar form have not been submitted to any examination body and have not been published. This paper was not yet, even in part, used in another examination or as a course performance. I agree that my work may be checked by a plagiarism checker.

12.01.2023&Lippstadt - Pritish Samant

Bibliography

[De11] Delahaye, Benoit; Katoen, Joost-Pieter; Larsen, Kim Guldstrand; Legay, Axel; Pedersen, Mikkel Larsen; Sher, Falak; Wasowski, Andrzej: New Results on Abstract Probabilistic Automata. IEEE Communications Society, pp. 118–127, 2011. Proceedings. Eleventh International Conference on Application of Concurrency to System Design (ACSD 2011). ISBN: 978-0-7695-4387-1.; null ; Conference date: 20-06-2011 Through 24-06-2011.

[dFC20] da F. Costa, Luciano: , Where Do Patterns To Be Recognized Come From? (CDT-22), 02 2020.

[KC02] Kuhn, D. Richard; Chandramouli, Ramaswamy: Cost Effective Use of Formal Methods in Verification and Validation. 2002.

[La17] Larsen, Kim Guldstrand: Validation, Synthesis and Optimization for Cyber-Physical Systems. In: TACAS. 2017.

[LWL14] Li, Qiang; Wang, Xiaoyan; Liu, Shufen: Quantitative model verification in VANET based on interval probabilistic timed automata. In: Proceedings of the 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design (CSCWD). pp. 418–422, 2014.

[MSW16] Muccini, Henry; Sharaf, Mohammad; Weyns, Danny: Self-Adaptation for Cyber-Physical Systems: A Systematic Literature Review. In: Proceedings of the 11th International Symposium on Software Engineering for Adaptive and Self-Managing Systems. SEAMS '16, Association for Computing Machinery, New York, NY, USA, p. 75–81, 2016.

[Sh11] Shi, Jianhua; Wan, Jiafu; Yan, Hehua; Suo, Hui: A survey of Cyber-Physical Systems. 2011 International Conference on Wireless Communications and Signal Processing (WCSP), pp. 1–6, 2011.

[St02] Stoelinga, Marielle: An Introduction to Probabilistic Automata. Bull. EATCS, 78:176–198, 2002.

[Ve21] Venkatraman, s; Muthusamy, P.; Balusa, Bhanuchander; Thangaiyan, Jayasankar; Kavithaa, G.; Sekar, Dr K R; Bharatiraja, C.: Time dependent anomaly detection system for smart environment using probabilistic timed automaton. Journal of Ambient Intelligence and Humanized Computing, pp. 1–9, 01 2021.

[ZJ15] Zheng, Xi; Julien, Christine: Verification and Validation in Cyber Physical Systems: Research Challenges and a Way Forward. In: Proceedings of the First International Workshop on Software Engineering for Smart Cyber-Physical Systems. SEsCPS '15. IEEE Press, p. 15–18, 2015.