Model and Verify CPS using "Markov Chains"

Pritish Samant[1]

Contents

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

Abstract:

In our day to day life, Cyber Physical System(CPS) plays a very important role. The range of Cyber Physical System includes from automation to self driving cars to upgradation of technologies. All the systems must be formally verified using some form of schotastic modeling to ensure the correctness. Markov Chains is one of the important part related to modeling and verification of Cyber Physical Systems(CPS). In this paper, we will be using Markov chains to model and verify the Cyber Physical System(CPS).

1 Motivation

In recent times, as there are technological advancements in this world, the one thing that has helped in this advancement and nurtured the growth is Cyber Physical System (CPS). "Systems that combine computing and physical processes are known as cyber-physical systems (CPS)" [Le08]. The intersection is smooth and in synchronization with real world systems. These systems have helped in real life technological advancements. In today's world, growth in technology has been attributed to Cyber Physical System. But it needs to be error free. And that is why, after modeling the system, a formal verification is required to check the system and make it error free.

Sensors, actuators, and controllers make up the three essential components of CPS [JR18]. Embedded System Consists of sensors and actuators whereas Controller belongs to cyber or software system. All these together need to work synchronously so that the complex system creates no errors. This is how the architecture of CPS looks like. For smaller systems, CPS might not look more useful but for bigger and complex systems which are mostly used for automation require a Cyber Physical system. And to check not only its correctness but also to make it more synchronous and reliable, we need to verify the system.

There are multiple ways to verify such a system. In this paper, we are going to talk about one of the method called as Markov Chains to verify a system. Markov chains is a stochastic model which determines the probability of a state or transitions. Markov Chains can be used for multiple use cases and real life situations. Such a stochastic model can be used for verifying Cyber Physical System. There are various tools to verify the property of a markov chain, which we will further see in this paper. Markov chains are one of the most sophisticated and a very important stochastic model that can used to verify and satisfy the property of a system. The process is called as Markov process and the property is markov property. So, the markov process that satisfies markov property is Markov Chains.

2 Foundation

2.1 Cyber Physical System

When monitoring and controlling processes are being carried out, the application of the CPS will help to overcome issues with credibility, real-time capability, and security levels [JR18].

Using CPS, we can ensure the systems are running smoothly and error free. As these systems are already complex and with continuous development these are getting more complex day by day, we are left we no choice other than to increase the reliability and safety of the system. Although it has become necessary to build CPS, there are some disadvantages or rather limitations in building these systems and to keep a check on such large and complex system. The economic factor also plays an important role in deciding the research and upgradation of systems. Since these systems have a direct effect on real human life and ongoing research, we cannot have any form of loopholes running around the system. Further in this paper, we will see, to avoid any mishap with the CPS, how we would be using Markov chains to model and verify the system.

2.2 Markov Chains

A conventional finite-state Markov chain is a stochastic process satisfying two main assumptions [FK10]:

A1. The future is independent of the past given present (Markov assumption)

A2. The set of possible values of the uncertain variables belongs to a finite set which is called the state space of the Markov chain.

We denote the state space of the finite state Markov chain as $S = \{\bar{x}_j, j = 1, \ldots, M\} \subset X$, where $X \subset \mathbb{R}$ is bounded. We introduce transition probabilities as $\pi_{ij} = P(x^+ = \bar{x}_j \,|\, x = \bar{x}_i)$, which form a transition probability matrix, Π. For a given $x = \bar{x}_i$, the next state inferred by the Markov chain is the one that maximizes the probability distribution corresponding to $\bar{x}_i$:

$$x^+ = \bar{x}_j, \text{ if } x = \bar{x}_i, \; j \in \arg\max_k \pi_{ik}$$

The markov models are named after Andrey Markov, a Russian mathematician who was born in 1856 and died in 1922[GCM22]. He studied Markov Process. Markov Chain is series of events and their probabilities. It says that the next event depends on current event and not the past. Now we will see an example of how Markov process works. Assume a person named Max. When he is at home, its his starting or initial location. And he has three locations where he can go named A,B,C. Now we will create a system for the same

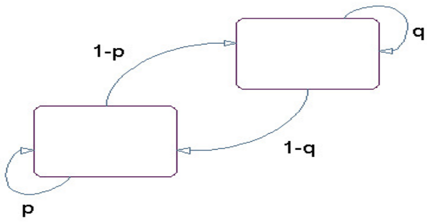

Fig. 1: Markov Chains [Ma]

which is also called as a Markov Model. In this mode, each location is a state. Every state has a probability and is called as Transition Probabilities. So, Every location Max goes is a state and the possibility of max going to that specific location out of A,B, or C decides the probabilities of the model. The total of all the probabilities of a model is always 1. As stated earlier, lets say Max is at home. Its his starting location. So, Max has 0.6 chance of going to location A and 0.3 chance of going to Location B and 0.1 chance of going to location C. This values are calculated based on Max's location and the number of times Max visits these locations. Which basically translates to Max left home 10 times, out of which he goes to location A, 6 times and location B, 3 times and location C, 1 time. All these probabilities constitute a Markov model. Its then represented as a transition matrix which has all the values of probabilities of all the states. This matrix can be used to calculate different unknowns from the system or a model. This is the simple explanation of Markov model and the Markov process using an example. Although this sounds simple, this model can be used as a powerful tool for a complex system. This is an important part of stochastic modelling.

3 Modeling of Cyber Physical System using Markov Chains

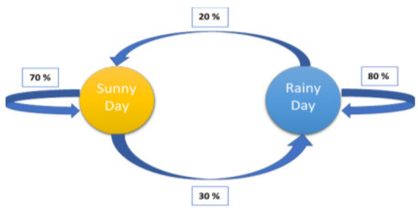

Fig. 2: weather example [Ma20b]

In the above figure, we can see that in a day, either its sunny or rainy. There are some assumptions we make. One being, the weather on present day depends on weather of last day and not before that or after that and second assumption being the weather of being sunny or raining depends on all the probabilities of the system. There are four probabilities of this system, Weather being sunny to rainy, sunny to sunny, rainy to sunny and rainy to rainy. In the above example, we can see that the probability of being sunny the next day is 0.7 and probability of weather being rainy next day is 0.3. Similarly, probability of being rainy from present rainy day is 0.8 and probability of being sunny next day if present day is rainy is 0.2. Comparing the above figure with markov chains, we have two states, sunny and rainy. In this example we just have two states but we can also have multiple states. This is a markov chain because it is based on markov assumptions. we discussed earlier about the assumptions. those same assumptions make this a markov chain. Then comes the transition probabilities. Its just a matrix. Its basically, all the probabilities of all transitions of states of a markov chain presenting as values in mathematical form. Using this, we can calculate different unknowns.

the mathematical equation is as[SC18]: state1 * transition matrix = new state

We can also use python script instead of manual markov chain. the python code follows[Ma20a]:

```
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt

transition_matrix = np.atleast_2d([[0.7, 0.3],
                                   [0.2, 0.8]])

possible_states = ['sunny', 'rainy']

start_state = 'sunny'
number_of_future_states_to_generate = 14
index_dict = {possible_states[index]: index for index in range(len(possible_states))}
future_states = []
for i in range(number_of_future_states_to_generate):
    new_state = np.random.choice(possible_states, p=transition_matrix[index_dict[start_state], :])
    future_states.append(new_state)
```

The result we get for the next five days is sunny, rainy, sunny, sunny, sunny.

4 Verification of Cyber Physical System using Markov Chains

Generally, A user decides the path and also the functioning of the system. For this, specifications of the System is important.These Specifications are designed by the user according to the system. Then this specification is converted into steps to perform. The specification and the steps to perform should be same. If these are not same, then the system is more susceptible to outside attacks. This process has to be done for every new system. To overcome this, a software is tool is used which makes sure system is going the right way and the conditions like steps to perform are met. This is a part of Formal Verification.

Markov chain can be formally verified using model checker tools like Prism, $EEMC^2$, Ymer, and Vesta. In this paper, we used the prism model checker tool. It offers immediate guidance for the analysis of three categories of probabilistic models: Markov decision processes (MDPs), discrete-time Markov chains (DTMCs), and continuous-time Markov chains (CTMCs) [KNP04]. Although for this, we used MDP.

Fig. 3: Prism tool

This is a screenshot of prism tool while running. Here we will see the prism code that describes the above markov chain example. We also used the prism tool for the verification of the model. Similarly, we can verify different cyber physical systems using markov chains.

```
mdp

// States
const int sunny;  // State for sunny weather
const int rainy;  // State for rainy weather

// Probabilities
const double sunnyToSunny = 0.7;  // Probability of staying sunny
const double sunnyToRainy = 0.3;  // Probability of changing to rainy
const double rainyToRainy = 0.8;  // Probability of staying rainy
const double rainyToSunny = 0.2;  // Probability of changing to sunny

module weather
  state: [0..1] init 0; // 0 represents sunny, 1 represents rainy

  [] state = sunny -> sunnyToSunny: (state' = sunny) + sunnyToRainy: (state' = rainy);
  [] state = rainy -> rainyToRainy: (state' = rainy) + rainyToSunny: (state' = sunny);
endmodule
```

This prism code defines two states, sunny and rainy. Then there are transition probabilities.

Here we have 4 transitions that predicts weather change. This code defines the markov chain model for the above weather example.

5 Conclusion

Using the above use case, we can see how the markov chains concept can be used to verify a system or a model. We have also used prism tool to verify the system or model. Similarly, we can use markov chains to interpret a probabilistic model and also verify the model. Markov chains although are very simple in concept but their application can be extended to some of the most complex cyber physical system. In short, markov chains are an important part of stochastic modelling and there are available resources to verify the correctness of the model as well.

6 Declaration of Originality

I, Pritish Samant, 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.

03.12.2023, Lippstadt - Pritish Samant. Pritish

Date&Place - Full Name

Bibliography

[FK10] Filev, Dimitar P.; Kolmanovsky, Ilya: A generalized Markov Chain modeling approach for on board applications. In: The 2010 International Joint Conference on Neural Networks (IJCNN). pp. 1–8, 2010.

[GCM22] Gao, Tianhan; Chen, Baicheng; Mi, Qingwei: A Survey of Markov Model in Reinforcement Learning. In: 2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC). pp. 284–287, 2022.

[JR18] Jamaludin, Juliza; Rohani, Jemmy Mohd: Cyber-Physical System (CPS): State of the Art. In: 2018 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube). pp. 1–5, 2018.

[KNP04] Kwiatkowska, M.; Norman, G.; Parker, D.: PRISM 2.0: a tool for probabilistic model checking. In: First International Conference on the Quantitative Evaluation of Systems, 2004. QEST 2004. Proceedings. pp. 322–323, 2004.

[Le08] Lee, Edward A.: Cyber Physical Systems: Design Challenges. In: 2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC). pp. 363–369, 2008.

[Ma] Markov Chains - MATLAB & Simulink - MathWorks Deutschland. https://de.mathworks.com/help/stats/markov-chains.html. Accessed: 2023-11-3.

[Ma20a] Markov chains. https://deparkes.co.uk/2020/08/08/markov-chains/, August 2020. Accessed: 2023-11-7.

[Ma20b] Markov Chains: What are they and where do they matter? https://www.latentview.com/blog/markov-chains-what-are-they-and-where-do-they-matter/, August 2020. Accessed: 2023-11-7.

[SC18] Sharma, Prateek; Chetty, Priya: , Markov chain and its use in solving real world problems. https://www.projectguru.in/markov-chain-real-world-problems/, February 2018. Accessed: 2023-11-7.