NOC:Model Checking - Video course

COURSE OUTLINE

Embedded software control many of the safety-critical systems that we deal with in everyday life: for instance, modern cars are equipped with software to automatically change gears; pacemakers come with a software controller to regulate heart beat; aircrafts have flight control software, and so on. Typically, these (software) controllers have to make decisions based on inputs coming from multiple interacting components. As the size and the number of interacting components increase, the design and verification of controllers becomes increasingly complex.

Model checking is a field of research that addresses this challenge by making use of mathematical models in the design and verification of controllers. The main idea is to look at the system as a mathematical model - commonly used models are extensions of finite-state machines. Design requirements on the controller then get translated to suitable questions on these mathematical models.The goal of this course is to understand some of the techniques and tools used in the process of model-checking.

COURSE DETAIL

Week. No	Topics
1.	Modeling systems as Finite-state machines
2	Using the model-checker NuSMV
3	Linear-time properties for verification
4	Regular properties – automata over finite words

NP-TEL

NPTEL

http://nptel.ac.in

Computer Science and Engineering

Pre-requisites:

Familiarity with basic algorithms and finite-state machines preferable

Coordinators:

Prof. B. Srivathsan
CMI

5	Omega-regular properties – automata over infinite words
6	Model checking omega-regular properties
7	Linear Temporal Logic (LTL)
8	Algorithms for LTL
9	Computation Tree Logic (CTL)
10	Algorithms for LTL
11	Models with timing constraints – timed automata
12	More on timed automata
13	Probabilistic models I
14	Probabilistic models II
15	Probabilistic models III

References:

Principles of Model-checking, *Christel Baier and Joost-Pieter Katoen*, MIT Press (2008)