Model checking: an introduction

FMSS, 2019-2020

Model checking

A fully automated method for analysing properties of systems.

A verification technique based on:

- model of the system described by a transition system
- computation tree logic for expressing properties
- automatic check that the model satisfies the formula

E.M. Clarke, E.A. Emerson and A.P. Sistla, Automatic verification of finite state concurrent systems using temporal logic specifications. ACM Transactions on Programming Languages and Systems, Vol. 8, No. 2, 1986.

Model checking

In case of programs:

- the control flow graph represents the structure of the program.

- transition systems represent information on the control structure plus some additional data.

- additional information assigned to states (e.g., the transition system generated by the operational semantics)

Transition System

A transition system TS is a tuple $(S, I, \rightarrow, AP, L)$ where:

- S is a non empty set of states;

- $I \subseteq S$ is a non-empty set of initial states;

$\rightarrow \subseteq S \times S$ is the transition relation;

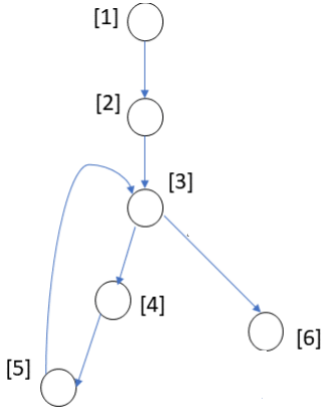
- AP is a set of atomic propositions;

$L : s \Rightarrow PowerSet(AP)$ is a labelling function for states.

We write $s \rightarrow s'$ if there is a transition from state s to state s'.

Transition System

$S = \{1, 2, 3, 4, 5, 6\}$
$I = \{1\}$
$\rightarrow$ in figure
$AP = \{a, b\}$
$L(1) = L(2) = L(3) = a$
$L(4) = L(5) = \{a, b\}$
$L(6) = \{b\}$



CTL formula: ϕ
The simplest formula is an atomic proposition.

We write:
$s \models \phi$
state s satisfies ϕ, i.e., state s is a model of ϕ

Example: assume $\phi = a$. $s \models \phi$ (a holds right now)

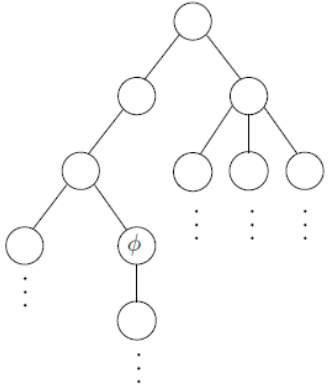
Reachability in one step

$EX\phi$ it is possible in one step to reach a state that satifies ϕ
$AX\phi$ the next state it is certain that satifies ϕ

Reachability

EF ϕ

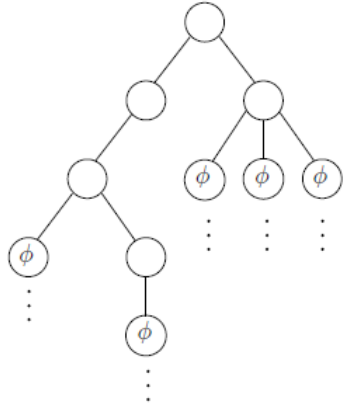
There exists a path and a state along that path such that ϕ holds



Reachability

AF ϕ

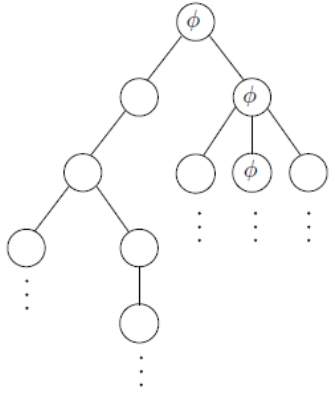
Along every path there exists a state such that ϕ holds at that state



Unavoidability

EG ϕ

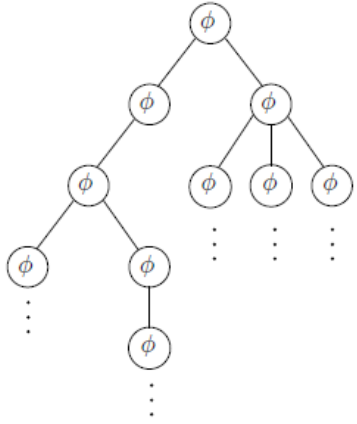
There exists a path such that ϕ holds at every state along that path



Unavoidability

AG ϕ

Along every path ϕ holds at every state



Computation Tree Logic

CTL

STATE FORMULAE

$$\phi ::= tt \mid ap \mid \phi_1 \wedge \phi_2 \mid \neg\phi \mid E\Psi \mid A\Psi$$

PATH FORMULAE

$$\Psi ::= X\phi \mid F\phi \mid G\phi \mid \phi_1 U \phi_2$$

U = until (ϕ_1 until ϕ_2)

Model checking

- Model checking works on a finite state model of the system. Verification is usually carried out by expressing a desired property of the system as a logic formula and then verifying by model checking algorithms that the model satisfies the formula.
- Model checking approach gives a direct automatic verification method of system properties.

Model checking

- Unfortunately, this approach has the drawback that systems composed of several subsystems can be associated a finite state model with a number of states which is exponential in the number of the component subsystems.
- Moreover, systems which are highly dependent on data values, share the same problem producing a number of states exponential in the number of data variables.
- This is the so called "State Space Explosion problem".