

# Introduction

Rodica Condurache

Lecture 1

## MODEL CHECKING: Is the system correct?

- **Reactive systems** are **non terminating, continuously interacting with the environment** for which the **safety is critical**.

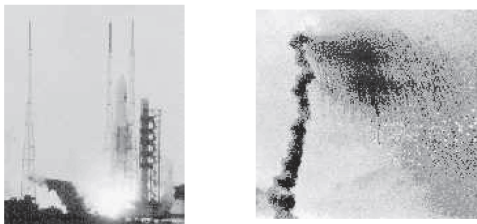


Figure 1.1: The Ariane-5 launch on June 4, 1996; it crashed 36 seconds after the launch due to a conversion of a 64-bit floating point into a 16-bit integer value.

- Example: Therac-25 caused the death of six cancer patients between 1985 and 1987 as they were exposed to an overdose of radiation.
- Other safety-critical systems : aschemical plants, nuclear power plants, traffic control and alert systems, and storm surge barriers

- **Program verification:** algorithmic methods to check whether a **system** satisfies a given specification .



$\models?$

Specification

Formal Methods

Automata  
Models

$\models?$

Temporal Logic

e.g. LTL

- The state : capture the essential features of the system under investigation, e.g. the security of a computer system.

# Model Checking process

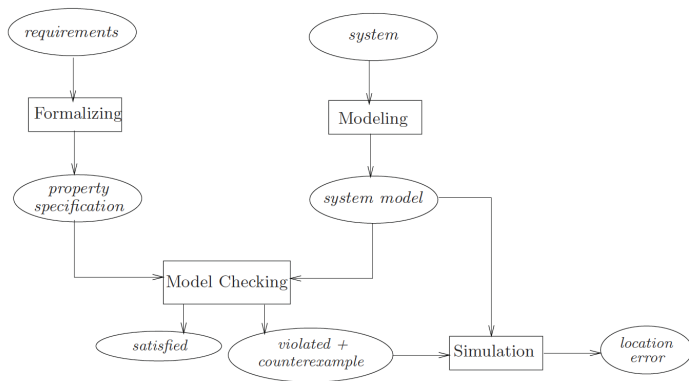


Figure 1.4: Schematic view of the model-checking approach.

# Model Checking process

- *Modeling* phase:
  - model the system under consideration using the model description language of the model checker at hand;
  - as a first sanity check and quick assessment of the model perform some simulations;
  - formalize the property to be checked using the property specification language.
- *Running* phase: run the model checker to check the validity of the property in the system model.
- *Analysis* phase:
  - property satisfied? → check next property (if any);
  - property violated? →
    1. analyze generated counterexample by simulation;
    2. refine the model, design, or property;
    3. repeat the entire procedure.
  - out of memory? → try to reduce the model and try again.

## MC - Strengths

- a **general verification approach** applicable to a wide range of applications : embedded systems, software engineering, and hardware design.
- It supports **partial verification** : properties can be checked individually
- It provides **diagnostic information** in case a property is invalidated
- It enjoys a rapidly increasing **interest by industry**;
  - Intel, AMD, NVIDIA, ARM : designul circuitelor integrate (cipurilor) este corect inainte de productia fizica (tools : Cadence JasperGold, Synopsys VC Formal)
  - Amazon Web Services : designul serviciilor de sticare si calcul distribuit
  - Industria Aerospatiala si Aviatica : Airbus, NASA
  - Industria Automotive : verificarea sistemelor de asistenta (ADAS)
  - Securitatea cibernetica

- mainly **appropriate to control-intensive applications** and less suited for data-intensive applications as data typically ranges over infinite domains.
- Its **applicability is subject to decidability issues** (for infinite-state systems, or reasoning about abstract data types)
- It **verifies a system model**, and not the actual system
- It **checks only stated requirements**, i.e., there is no guarantee of completeness. The validity of properties that are not checked cannot be judged.
- It suffers from the **state-space explosion problem**,
- **requires some expertise** in finding appropriate abstractions to obtain smaller system models and to state properties

## SYNTHESIS PROBLEM: generate a correct system

- Designing a system can be hard.

*Any verification using model-based techniques is only as good as the model of the system.*

- **Synthesis problem:** automatically generate the system from the specification.

?  $\models$  **Specification**

- Difficulty: the environment is **uncontrollable**.
- Synthesis  $\approx$  a **two-player game** between an environment and a system.
- **State of the art:**
  - Introduced long time ago (Church)
  - New interest in **feasible methods for the synthesis** of:
    - reactive systems ,
    - distributed systems ,
    - programs manipulating bit streams ,
    - arithmetic
    - concurrent data-structures.

# Modal logic

---

Rodica Condurache

Lecture 1

# Propositional Logic

**Propositional Logic** contain formulas over a set of atomic propositions  $\mathcal{AP}$

## Definition (LP syntax)

Given a set  $\mathcal{AP}$  of atomic propositions, an PL formula over  $\mathcal{AP}$  is defined by the following syntax:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi$$

where  $p \in \mathcal{AP}$ .

- We can define the following macros:

$$\varphi_1 \wedge \varphi_2 = \neg(\neg\varphi_1 \vee \neg\varphi_2) \quad (\varphi_1 \text{ and } \varphi_2)$$

$$\varphi_1 \rightarrow \varphi_2 = \neg\varphi_1 \vee \varphi_2 \quad (\varphi_1 \text{ implies } \varphi_2)$$

$$\varphi_1 \leftrightarrow \varphi_2 = (\varphi_1 \rightarrow \varphi_2) \wedge (\varphi_2 \rightarrow \varphi_1) \quad (\varphi_1 \text{ equivalent to } \varphi_2)$$

- In Propositional Logic, we only can say that a property is **true** or **false**.
- We would like to express things like:
  - "The property  $\varphi$  is **necessary true**"
  - "The property  $\varphi$  **may be true**"
  - "The property  $\varphi$  is **always true in the future**"
  - "The property  $\varphi$  **eventually becomes true in the future**"

## Modal Logics - History

- Modal logic was first discussed in a systematic way by Aristotle

*" $p$  is possible may be defined as  $not-p$  is not necessary."*

- C.I.Lewis - founded the modern modal logic
  - several systems to characterize the logical consequence relation;
- R. Carnap - Instead of considering modal propositions, he considered modal sentences and evaluated them in state descriptions.
  - State descriptions = sets of simple (atomic) sentences
  - The simple sentence  $p$  is true with respect to a state description  $S$  iff  $p \in S$ .

*"Necessarily  $p$  is true in  $S$  iff for every state-description  $S'$  in  $M$ ,  $p$  is true in  $S'$ ."*

- Kripke introduced a domain of possible worlds and regarded the modal prefix 'it is necessary that' as a quantifier over worlds

*"Necessarily  $p$  is true at a world  $w$  iff  $p$  is true at every world  $w'$  accessible from  $w$ ."*

- Amir Pnueli - proposed that temporal logic should be used for checking continually operating concurrent programs

## Basic Modal Logic - Syntax

- We define a class of very general modal languages over a set of atomic propositions  $\mathcal{AP}$ .

### Definition (Basic Modal Logic - syntax)

Given a set  $\mathcal{AP}$  of atomic propositions, an BML formula over  $\mathcal{AP}$  is defined by the following syntax:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \Box\varphi \mid \Diamond\varphi$$

where  $p \in \mathcal{AP}$ .

We read :

- $\Box\varphi$  as 'necessarily  $\varphi$ '
- $\Diamond\varphi$  as 'possibly  $\varphi$ '

## Basic Modal Logic - Kripke Structures

- Formulas in BML are evaluated over Kripke structures

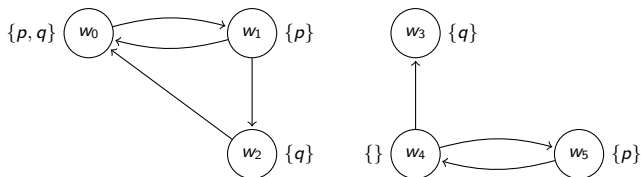
### Definition (Kripke Structures)

A standard model  $\mathcal{M}$  for a set of atomic formulas  $\mathcal{AP}$  is a tuple  $\langle W, R, L \rangle$  where

- $W$  is a non-empty set (the set of worlds)
- $R \subseteq W \times W$  is the accessibility relation between worlds
- $L : W \rightarrow 2^{\mathcal{AP}}$  is the valuation (labeling) function

## Kripke Structure - Example

- $W = \{w_0, w_1, w_2, w_3, w_4, w_5\}$ ,
- $R = \{(w_0, w_1), (w_1, w_0), (w_1, w_2), (w_2, w_0), (w_4, w_5), (w_5, w_4), (w_4, w_3)\}$
- $L(w_0) = \{p, q\}$ ,  $L(w_1) = \{p\}$ ,  $L(w_2) = \{q\}$ ,  $L(w_3) = \{q\}$ ,  $L(w_4) = \emptyset$ ,  $L(w_5) = \{p\}$ .



## Basic Modal Logic - Semantics

BML formulas are evaluated over worlds in a Kripke structure:

### Definition (Basic Modal Logic - semantics)

Given a Kripke structure  $\mathcal{M} = \langle W, R, L \rangle$ , a world  $w \in W$  and two BML formulas  $\varphi$  and  $\psi$ , we have:

- $\mathcal{M}, w \models p$  iff  $p \in L(w)$ ,
- $\mathcal{M}, w \models \neg\varphi$  iff  $\mathcal{M}, w \not\models \varphi$ ,
- $\mathcal{M}, w \models \varphi \vee \psi$  iff  $\mathcal{M}, w \models \varphi$  or  $\mathcal{M}, w \models \psi$ ,
- $\mathcal{M}, w \models \Box\varphi$  iff for each  $t \in W$  with  $(w, t) \in R$ , we have  $\mathcal{M}, t \models \varphi$ ,
- $\mathcal{M}, w \models \Diamond\varphi$  iff there exists  $t \in W$  s.t.  $(w, t) \in R$  and  $\mathcal{M}, t \models \varphi$ .

**Intuition:** If  $(w, w') \in R$ , then the propositions true at  $w'$  are possible at  $w$ .

## Basic Modal Logic - Semantics

BML formulas are evaluated over worlds in a Kripke structure:

### Definition (Basic Modal Logic - semantics)

Given a Kripke structure  $\mathcal{M} = \langle W, R, L \rangle$ , a world  $w \in W$  and two BML formulas  $\varphi$  and  $\psi$ , we have:

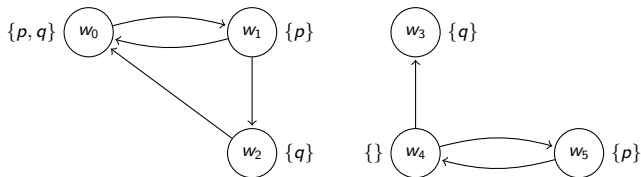
- $\mathcal{M}, w \models p$  iff  $p \in L(w)$ ,
- $\mathcal{M}, w \models \neg\varphi$  iff  $\mathcal{M}, w \not\models \varphi$ ,
- $\mathcal{M}, w \models \varphi \vee \psi$  iff  $\mathcal{M}, w \models \varphi$  or  $\mathcal{M}, w \models \psi$ ,
- $\mathcal{M}, w \models \Box\varphi$  iff for each  $t \in W$  with  $(w, t) \in R$ , we have  $\mathcal{M}, t \models \varphi$ ,
- $\mathcal{M}, w \models \Diamond\varphi$  iff there exists  $t \in W$  s.t.  $(w, t) \in R$  and  $\mathcal{M}, t \models \varphi$ .

**Intuition:** If  $(w, w') \in R$ , then the propositions true at  $w'$  are possible at  $w$ .

We say that  $\varphi$  is *globally true* and write  $\mathcal{M} \models \varphi$  iff  $\mathcal{M}, w \models \varphi$  for all  $w \in W$ .

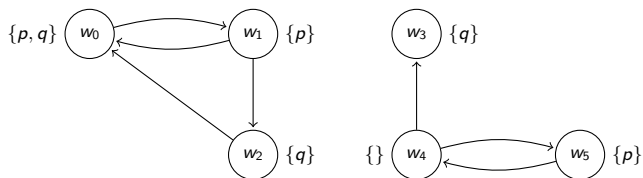
The formula  $\varphi$  is *satisfiable* in  $\mathcal{M}$  iff there exists  $w \in W$  s.t.  $\mathcal{M}, w \models \varphi$ .

## Example



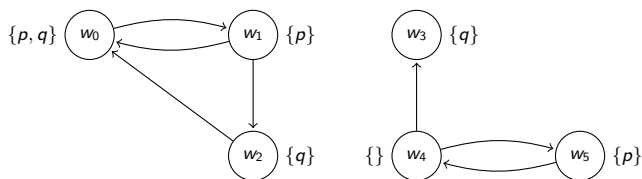
- $\mathcal{M}, w_2 \models q$ ,
- $\mathcal{M}, w_2 \not\models p$
- $\mathcal{M}, w_1 \models \diamond p$
- $\mathcal{M}, w_1 \not\models \Box p$
  
- $\mathcal{M}, w_4 \not\models \Box q$
- $\mathcal{M}, w_4 \not\models \Box p$
- $\mathcal{M}, w_4 \not\models (\Box q) \vee (\Box p)$
- but  $\mathcal{M}, w_4 \models \Box(p \vee q)$

## Example



- In which worlds holds  $\Diamond T$ ?

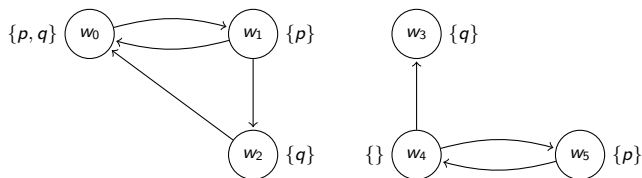
## Example



- In which worlds holds  $\Diamond T$ ?

Even though all the worlds in a Kripke model satisfy  $T$ , there are worlds where  $\Diamond T$  is false ! (all except  $w_3$ )

## Example

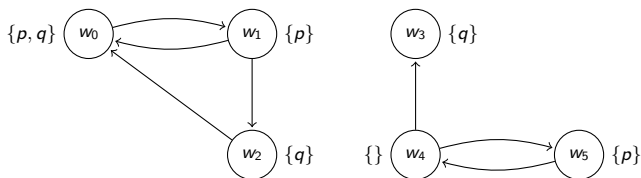


- In which worlds holds  $\Diamond \top$ ?

Even though all the worlds in a Kripke model satisfy  $\top$ , there are worlds where  $\Diamond \top$  is false ! (all except  $w_3$ )

- In which worlds holds  $\Box \perp$ ?

## Example



- In which worlds holds  $\Diamond \top$ ?

Even though all the worlds in a Kripke model satisfy  $\top$ , there are worlds where  $\Diamond \top$  is false ! (all except  $w_3$ )

- In which worlds holds  $\Box \perp$ ?

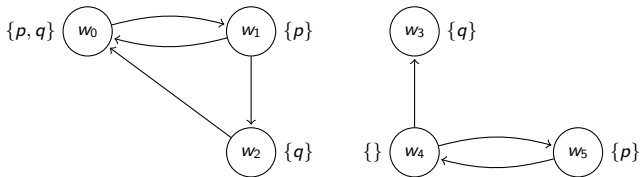
Even though none of the worlds in a Kripke structure satisfies  $\perp$ , there are worlds where  $\Box \perp$  is true ! (see  $w_3$ )

## Definition

A model  $\mathcal{M} = \langle W, R, L \rangle$  is said to **satisfy a formula**  $\varphi$  if every state in the model satisfies it. Thus, we write  $\mathcal{M} \models \varphi$  if and only if  $\mathcal{M}, w \models \varphi$  for every  $w \in W$ .

## Definition

A model  $\mathcal{M} = \langle W, R, L \rangle$  is said to **satisfy a formula**  $\varphi$  if every state in the model satisfies it. Thus, we write  $\mathcal{M} \models \varphi$  if and only if  $\mathcal{M}, w \models \varphi$  for every  $w \in W$ .



Let  $\varphi = q \rightarrow \Box p$

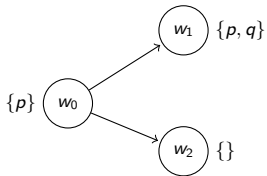
Is it true that  $\mathcal{M} \models \varphi$ ?

## Example 2

- Do the formulas  $\Box(p \rightarrow q)$  and  $p \rightarrow \Box q$  have the same truth condition?

## Example 2

- Do the formulas  $\Box(p \rightarrow q)$  and  $p \rightarrow \Box q$  have the same truth condition?



### Definition (Equivalent Formulas)

We say that two formulas  $\varphi$  and  $\psi$  are semantically **equivalent** if for any Kripke structure  $\mathcal{M}$  and any world  $w$  in  $\mathcal{M}$ , holds

$$\mathcal{M}, w \models \varphi \text{ iff } \mathcal{M}, w \models \psi$$

- Prove that  $\diamond\varphi \equiv \neg\Box\neg\varphi$

## Equivalent and Valid Formulas

- Intuitively,  $\Box$  behaves like the universal quantifier while  $\Diamond$  behaves like the existential quantifier.
- Likewise  $\exists$  and  $\forall$ , modal operators  $\Diamond$  and  $\Box$  distributes over  $\vee$  and  $\wedge$ , respectively.
  - $\Box(\varphi \wedge \psi) \equiv (\Box\varphi) \wedge (\Box\psi)$
  - $\Diamond(\varphi \vee \psi) \equiv (\Diamond\varphi) \vee (\Diamond\psi)$

### Definition (Valid formulas)

A formula of BML is said to be **valid** if it is true in any world  $x \in W$  of every model  $\mathcal{M} = \langle W, R, L \rangle$ .

- by replacing  $\equiv$  by  $\leftrightarrow$  in the above equivalences, we obtain valid formulas.
  - $\Box(\varphi \wedge \psi) \leftrightarrow (\Box\varphi) \wedge (\Box\psi)$  and
  - $\Diamond(\varphi \vee \psi) \leftrightarrow (\Diamond\varphi) \vee (\Diamond\psi)$  are valid

## Exercise 1

Consider the following model:  $\mathcal{M} = \langle W, R, L \rangle$  where:  $W = \{a, b, c, d, e\}$ ,  $R = \{(a, b), (a, e), (b, c), (b, e), (d, d), (e, e)\}$  and  $L(a) = \{p, q\}$ ,  $L(b) = \{p\}$ ,  $L(c) = \{p, q\}$ ,  $L(d) = \{q\}$ ,  $L(e) = \emptyset$ .

Determine if:

- $a \models p$
- $a \models \Box \neg q$
- $a \models \Box \Box q$
- $a \models \Diamond p$
- $a \models \Box \Diamond \neg q$
- $a \models \Diamond \Diamond (p \wedge q) \wedge \Diamond \top$

## Exercise 2

For the model described at Exercise 1, find a world that satisfy and a world that does not satisfy the following formulas:

- $\Box\neg p \wedge \Box\Box\neg p$ ,
- $\Diamond q \wedge \neg\Box q$
- $\Diamond p \vee \Diamond q$
- $\Diamond(p \vee \Diamond q)$
- $\Box p \vee \Box\neg p$
- $\Box(p \vee \neg p)$

## Exercise 3

For each pair of formulas, find a Kripke model that satisfy only one of them:

- $\Box p$  and  $\Box\Box p$
- $\Box\neg p$  and  $\neg\Diamond p$
- $\Box(p \vee q)$  and  $\Box p \vee \Box q$
- $\Diamond(p \wedge q)$  and  $(\Diamond p) \vee (\Diamond q)$
- $\Box(p \rightarrow q)$  and  $(\Box p) \rightarrow (\Box q)$

## Exercise 4

Prove that the following formulas are valid:

- $\Box p \rightarrow (\Box q \rightarrow \Box p)$
- $\Box(\varphi \wedge \psi) \rightarrow (\Box\varphi \rightarrow \Box\psi)$