The exam is composed of questions than can be answered in any order (assuming the results of previous questions if necessary). If a question is ambiguous or imprecise, it is then part of that question to solve the ambiguity or imprecision. The difficulty of each question is roughly estimated by a number of stars, from one for the easiest questions to three for the most difficult ones. Documents and computers are accepted.

1. LTL model-checking

Model-checking consists in verifying that a transition system (or Kripke structure) is a model of a temporal logic formula (LTL, CTL, CTL*, etc). It is easily shown, thanks to the following questions, that this is an abstract interpretation [1].

1.1 Models

A model or transition system or Kripke structure is a quadruple $M = \langle \Sigma, I, t, L \rangle$ where $\Sigma$ is a set of states, $I \subseteq \Sigma$ is a set of initial states, $t \subseteq \Sigma \times \Sigma$ is a transition relation between a state and its possible successors and $L \in \Sigma \mapsto \wp(\varphi(\Sigma))$ is a labeling of states by a set of atomic predicates chosen in a given set $\varphi(\Sigma)$ (with the interpretation that $p \in L(s)$ if and only if predicate $p$ holds in state $s$). The model is finite when $\Sigma$ and $\varphi(\Sigma)$ are finite.

We assume, as usual in model checking, that $t$ is total so that any state has at least one possible successor, formally $\forall s \in \Sigma : \exists s' \in \Sigma : \langle s, s' \rangle \in t$.

1.2 Paths

Let $\pi = \pi_0\pi_1...\pi_n... \in \Sigma^\omega$ be a path (or trace or trajectory), that is an infinite sequence $\pi \in \omega \mapsto \Sigma$ of states $\pi_n$, $n \geq 0$ in $\Sigma$. We write $\pi^k = \pi_k\pi_{k+1}...\pi_n...$ for the suffix of $\pi$ at rank $k$. In particular $\pi^0 = \pi$ and $\pi^1 = \pi_1\pi_2...\pi_n...$. We write $t^\omega$ for the set of paths of $t$, that is to say:

$$t^\omega = \{ \pi \in \Sigma^\omega | \forall i \geq 0 : \langle \pi_i, \pi_{i+1} \rangle \in t \}$$

We write $\text{lp}^\preceq f$ (respectively $\text{gfp}^\preceq f$) for the least (resp. the greatest) fixpoint of $f$ for the partial order $\preceq$, if any.

**Question 1.1 (⋆)** Given a model $M = \langle \Sigma, I, t, L \rangle$, characterize $t^\omega$ as a fixpoint.
1.3 LTL (syntax and semantics)

The formulæ $f$ of Amir Pnueli’s temporal logic LTL [3] are given as follows:

$$
p \in AP
$$

$$f ::= p \mid \neg f \mid f_1 \lor f_2 \mid Xf \mid f_1 U f_2 \mid Gf$$

We define the semantics of LTL as the subset of paths of $\Sigma^\omega$ for which a formula $f$ of LTL is true:

$$\llbracket f \rrbracket \in LTL \mapsto \varphi(\Sigma^\omega)$$

$$\llbracket p \rrbracket \triangleq \{ \pi \in \Sigma^\omega \mid p \in L(\pi_0) \}$$

$$\llbracket \neg f \rrbracket \triangleq \Sigma^\omega \setminus \llbracket f \rrbracket$$

$$\llbracket f_1 \lor f_2 \rrbracket \triangleq \llbracket f_1 \rrbracket \cup \llbracket f_2 \rrbracket$$

$$\llbracket Xf \rrbracket \triangleq \{ \pi \in \Sigma^\omega \mid \pi^1 \in \llbracket f \rrbracket \}$$

$$\llbracket f_1 U f_2 \rrbracket \triangleq \{ \pi \in \Sigma^\omega \mid \exists k \geq 0 : \pi^k \in \llbracket f_2 \rrbracket \land \forall i : (0 \leq i < k) \Rightarrow \pi^i \in \llbracket f_1 \rrbracket \}$$

$$\llbracket Gf \rrbracket \triangleq \{ \pi \in \Sigma^\omega \mid \forall i \geq 0 : \pi^i \in \llbracket f \rrbracket \}$$

**Question 1.2 (⋆)** Prove that:

$$\llbracket f_1 U f_2 \rrbracket = \text{lfp} \{ F[f_1, f_2] \}$$

where $F[f_1, f_2](X) \triangleq \llbracket f_2 \rrbracket \cup \{ \pi \in \llbracket f_1 \rrbracket \mid \pi^1 \in X \}$

**Question 1.3 (⋆⋆)** Characterize $\llbracket Gf \rrbracket$ as a fixpoint.

1.4 Classical semantics of LTL

The classical semantics of LTL [2] is not defined as we did in Sec. 1.3, but instead as the set of paths $\pi \in t^\omega$ of a model $M = \langle \Sigma, I, t, L \rangle$ which satisfy an LTL formula $f$. The classical definition is the following:

$$M, \pi \models p \triangleq p \in L(\pi_0)$$

$$M, \pi \models \neg f \triangleq M, \pi \not\models f$$

$$M, \pi \models f_1 \lor f_2 \triangleq M, \pi \models f_1 \text{ or } M, \pi \models f_2$$

$$M, \pi \models Xf \triangleq M, \pi^1 \models f$$

$$M, \pi \models f_1 U f_2 \triangleq \exists k \geq 0 : M, \pi^k \models f_2 \land \forall i : (0 \leq i < k) \Rightarrow M, \pi^i \models f_1$$

$$M, \pi \models Gf \triangleq \forall j \geq 0 : M, \pi^j \models f$$

**Question 1.4 (⋆)** Prove that for any LTL formula $f$ and any path $\pi \in t^\omega$ of the model $M = \langle \Sigma, I, t, L \rangle$, we have:

$$M, \pi \models f \iff \pi \in \llbracket f \rrbracket.$$
1.5 Abstraction

Given a model $M = \langle \Sigma, I, t, L \rangle$, we consider the abstraction:

$$
\alpha_M \in \varphi(\Sigma^\omega) \mapsto \varphi(\Sigma)
$$

$$
\alpha_M(X) \triangleq \{ \pi_0 \mid \pi \in X \cap t^\omega \}
$$

**Question 1.5 \((\star)\)** Prove that $\alpha_M$ is a surjective Galois connection:

$$
\langle \varphi(\Sigma^\omega), \subseteq \rangle \overrightarrow{\alpha_M} \langle \varphi(\Sigma), \subseteq \rangle
$$

**Question 1.6 \((\star)\)** Prove that $\alpha_M$ is a complete meet ($\cap$) morphism.

1.6 Model checking

Let us define $s\pi = \pi'$ such that $\pi'_0 = s$ and $\forall i \geq 0 : \pi'_{i+1} = \pi_i$. Checking of formula $f$ for a model $M = \langle \Sigma, I, t, L \rangle$ consists in verifying that:

— Existential verification:

$$
\exists s \in I : \exists s\pi \in t^\omega : s\pi \in [f]
$$

— Universal verification:

$$
\forall s \in I : \forall \pi \in \Sigma^\omega : s\pi \in t^\omega \land s\pi \notin [f]
$$

The two verifications derive from one another since:

$$
\forall s \in I : \exists \pi \in \Sigma^\omega : s\pi \in t^\omega \land s\pi \notin [f]
\Leftrightarrow \forall s \in I : \forall \pi \in \Sigma^\omega : s\pi \notin t^\omega \lor s\pi \in [f]
\Leftrightarrow \forall s \in I : \forall \pi \in \Sigma^\omega : s\pi \notin t^\omega \lor s\pi \notin [\neg f]
\Leftrightarrow \neg (\exists s \in I : \exists \pi \in \Sigma^\omega : s\pi \in t^\omega \land s\pi \in [\neg f])
$$

So we choose to study existential verification:

$$
\exists s \in I : \exists s\pi \in t^\omega : s\pi \in [f]
\Leftrightarrow I \cap \{ s \in \Sigma \mid \exists \pi \in \Sigma^\omega : s\pi \in t^\omega \land s\pi \in [f] \} \neq \emptyset
\Leftrightarrow I \cap \{ s \in \Sigma \mid \exists \pi \in \Sigma^\omega : s\pi \in [f] \cap t^\omega \} \neq \emptyset
\Leftrightarrow I \cap \{ \pi_0 \mid \pi \in [f] \cap t^\omega \} \neq \emptyset
\Leftrightarrow I \cap \alpha_M([f]) \neq \emptyset
$$

so that universal verification will be:

$$
\neg (I \cap \alpha_M([\neg f]) \neq \emptyset)
\Leftrightarrow I \cap \alpha_M([\neg f]) = \emptyset
\Leftrightarrow I \subseteq \neg \alpha_M([\neg f])
\Leftrightarrow I \subseteq \neg \alpha_M([-f])
$$
We don’t know\), one can start by computing \(\alpha(\llbracket f \rrbracket)\) before checking that \(I \cap \alpha_M(\llbracket f \rrbracket) \neq \emptyset\). We let:

\[
\text{pre}[t]X \triangleq \{ s \in \Sigma \mid \exists s' \in X : (s, s') \in t \}.
\]

Existential model checking, that is essentially the computation of \(\alpha_M(\llbracket f \rrbracket)\) can be done by the following algorithm (the iterative fixpoint computation terminating under the finiteness hypothesis):

**Question 1.7 \((**\)** Prove by induction on the syntax of \(f\) and abstraction that:

\[
\alpha_M(\llbracket p \rrbracket) = \{ s \in \Sigma \mid p \in L(s) \}
\]

\[
\alpha_M(\llbracket \neg f \rrbracket) = \neg \alpha_M(\llbracket f \rrbracket)
\]

\[
\alpha_M(\llbracket f_1 \lor f_2 \rrbracket) = \alpha_M(\llbracket f_1 \rrbracket) \cup \alpha_M(\llbracket f_2 \rrbracket)
\]

\[
\alpha_M(\llbracket \exists \phi \rrbracket) = \text{pre}[t](\alpha_M(\llbracket f \rrbracket))
\]

\[
\alpha_M(\llbracket f_1 \lor f_2 \rrbracket) = \text{lfp} \alpha \exists \chi(\alpha_M(\llbracket f_2 \rrbracket) \cup (\alpha_M(\llbracket f_1 \rrbracket) \cap \text{pre}[t](X)))
\]

\[
\alpha_M(\llbracket G \phi \rrbracket) = \text{gfp} \alpha \exists \chi(\alpha_M(\llbracket f \rrbracket) \cap \text{pre}[t](X))
\]

where \( \neg X \triangleq \Sigma \setminus X \) and \( \tilde{\alpha}_M(X) \triangleq \neg \alpha_M(\neg X) \) for which \( \tilde{\alpha}_M(\llbracket f \rrbracket) \) will be calculated by structural induction on \( f \).

## 2. Model checking for CTL and CTL∗

We now consider Allen Emerson’s temporal logic CTL∗ [2] which syntax is the following:

\[
p \in AP \quad \text{atomic formulæ}
\]

\[
f ::= p | \neg f | f_1 \lor f_2 | E[\phi] \quad \text{state formulæ}
\]

\[
\phi ::= f | \neg \phi | \phi_1 \lor \phi_2 | X\phi | \phi_1 U \phi_2 | G\phi \quad \text{path formulæ}
\]

Classically, the satisfaction relation for a CTL∗ formula and a model \( M = \langle \Sigma, I, t, L \rangle \) is defined as follows \((s \in \Sigma, \pi \in \Sigma^\omega)\):

\[
M, s \models p \triangleq p \in L(s)
\]

\[
M, s \models \neg f \triangleq M, s \not\models f
\]

\[
M, s \models f_1 \lor f_2 \triangleq M, s \models f_1 \text{ or } M, s \models f_2
\]

\[
M, s \models E[\phi] \triangleq \exists \pi \in t^\omega : s = \pi_0 \land M, \pi \models \phi
\]

\[
M, \pi \models f \triangleq M, \pi_0 \models f
\]

\[
M, \pi \models \neg \phi \triangleq M, \pi \not\models \phi
\]

\[
M, \pi \models \phi_1 \lor \phi_2 \triangleq M, \pi \models \phi_1 \text{ or } M, \pi \models \phi_2
\]

\[
M, \pi \models X\phi \triangleq M, \pi^1 \models \phi
\]

\[
M, \pi \models \phi_1 U \phi_2 \triangleq \exists k \geq 0 : M, \pi^k \models \phi_2 \land \forall i : (0 \leq j < k) \Rightarrow M, \pi^j \models \phi_1
\]

\[
M, \pi \models G\phi \triangleq \forall j \geq 0 : M, \pi^j \models \phi
\]
CTL is the subset of CTL\(^*\) obtained by using only \(\neg\), \(\lor\), \(E[Xf]\), \(E[f_1 \mathbf{U} f_2]\) and \(E[Gf]\) where \(f\), \(f_1\) and \(f_2\) are state formulæ:

\[
\begin{align*}
p & \in AP \quad \text{atomic formulæ} \\
f & ::= p \mid \neg f \mid f_1 \lor f_2 \mid E[\phi] \quad \text{state formulæ} \\
\phi & ::= Xf \mid f_1 \mathbf{U} f_2 \mid Gf \quad \text{path formulæ}
\end{align*}
\]

**Question 2.1 (★★★)** Provide a structural fixpoint algorithm to verify existentially a model \(M = \langle \Sigma, I, t, L \rangle\) for a state formula \(f\) of CTL:

\[
I \cap \{s \mid M, s \models f\} \neq \emptyset
\]

(or else universal verification, if preferred) by abstract interpretation. The answer should be inspired by Sec. 1., in particular by question 1.7

**Question 2.2 (★★★)** Taking inspiration from Sec. 2.1, do the same for CTL\(^*\).

**References**

