Home Assignment 1: Safety and Liveness

To hand in before or on October 26, 2009.
The penalty for delays is 2 points per day.

Electronic versions (PDF only) can be sent by email to ⟨schmitz@lsv.ens-cachan.fr⟩, paper versions should be handed in on the 26th or put in my mailbox at LSV, ENS Cachan.

This homework investigates the distinction between safety and liveness properties on infinite words. Informally, the definition for safety is that nothing “bad” (like a crash or a deadlock) ever happens, and for liveness that something “good” (like entering a critical section) eventually occurs. Let us consider a more concrete example to illustrate these notions:

Exercise 1 (A Mutual Exclusion Protocol). The following program is a mutual exclusion protocol for two processes due to Pnueli. There is a shared boolean variable $s$, initialized to 1, and two shared boolean variables $y_i$, $i$ in \{0, 1\}, initialized to 0. Each process $P_i$ can read the values of $s$, $y_0$, and $y_1$, but only write a new value in $s$ and $y_i$. Here is the code of process $P_i$ in C-like syntax:

```c
while (true)
{
    /* 1: Noncritical section. */
    atomic { $y_i = 1; s = i; \}$;
    /* 2: Wait for turn. */
    wait until (($y_1 - i == 0) \lor (s != i)));
    /* 3: Critical section. */
    $y_i = 0$;
}
```

1. Draw the transition system of each process, and construct their parallel composition. Label the states appropriately using the atomic propositions $w_i$ and $c_i$, holding when process $P_i$ is waiting or in the critical section, respectively.
2. Does the algorithm ensure *mutual exclusion*, i.e. that the two processes can never be simultaneously inside the critical section?

3. Give an LTL formula for mutual exclusion, i.e. such that all its models are traces where the two processes are never simultaneously inside the critical section.

4. Does the algorithm ensure *starvation freedom*, i.e. that every waiting process will eventually access the critical section, provided that the other process does not stay forever inside the critical section?

5. Give an LTL formula for starvation freedom.

The two mentioned properties, mutual exclusion and starvation freedom, are respectively a safety and a liveness property.

1 Topological Characterization

We consider as usual a finite alphabet $\Sigma$, the sets of finite words $\Sigma^*$ and of infinite words $\Sigma^\omega$, and $\Sigma^\infty = \Sigma^* \cup \Sigma^\omega$. This section provides a formal definition of safety and liveness properties, using some simple topological characterizations.

**Exercise 2** (Cantor Topology). Let us define a subset $O$ of $\Sigma^\omega$ as open if it is of form $W \cdot \Sigma^\omega$ with $W \subseteq \Sigma^*$. A closed subset is the complement of an open subset. A dense subset $D$ is such that the only closed subset of $\Sigma^\omega$ containing $D$ is $\Sigma^\omega$ itself. We further define the set of finite prefixes of a language $L$ included in $\Sigma^\omega$ as

$$
\text{Pref}(L) = \bigcup_{\sigma \in L} \text{Pref}(\sigma)
$$

$$
\text{Pref}(\sigma) = \{w \in \Sigma^* \mid \exists \sigma' \in \Sigma^\omega, \sigma = w\sigma'\}
$$

and the closure of a language $L$ included in $\Sigma^\omega$ as the language

$$
\text{cl}(L) = \{\sigma \in \Sigma^\omega \mid \text{Pref}(\sigma) \subseteq \text{Pref}(L)\} .
$$

1. Show that $\Sigma^\omega$, $\emptyset$, $\bigcup_i O_i$ for open sets $O_i$, and $\bigcap_i O_i$ for finitely many open sets $O_i$ are all open sets.

2. Show that $\text{cl}(L)$ is the smallest closed set containing $L$.

3. Show that $D \subseteq \Sigma^\omega$ is dense if and only if $\text{Pref}(D) = \Sigma^*$.

**Exercise 3** (Decomposition into Safety and Liveness). Recall that a property over a set of atomic propositions $\text{AP}$ is a language over $\Sigma = 2^{\text{AP}}$, i.e. a subset of $\Sigma^\omega$. A system verifies a property if its set of labeled traces is included in this language. Let us define a safety property as a closed subset of $\Sigma^\omega$, and a liveness property as a dense subset of $\Sigma^\omega$. The informal intuition behind these definitions is that
• if a safety property does not hold, then some “bad” behavior should occur at some point, thus after a finite time;
• no partial execution is irremediable for a liveness property: it always remains possible for the hoped for “good” behavior to occur at some future time.

1. Do the two properties studied in Exercise 1 comply with this formalization of safety and liveness: justify whether your LTL formulæ define closed or dense sets.

2. Prove the following theorem:

**Theorem 1** (Decomposition). For any property \( L \subseteq \Sigma^\omega \), there exist a safety property \( L_s \subseteq \Sigma^\omega \) and a liveness property \( L_l \subseteq \Sigma^\omega \) such that \( L = L_s \cap L_l \).

## 2 Past and Future LTL Formulæ

Recall that an \( \text{LTL}(Y, S, X, U) \) formula \( \varphi \) defines an aperiodic language \( L(\varphi) \):

\[
L(\varphi) = \{ \sigma = a_0a_1a_2 \cdots \in \Sigma^\omega \mid \sigma, 0 \models \varphi \}
\]

and that conversely, any aperiodic language included in \( \Sigma^\infty \) can be given a pure future formula. Note that these results include the case of aperiodic languages of finite words in \( \Sigma^+ \), where one defines

\[
L(\varphi) = \{ \sigma = a_0a_1 \cdots a_n \in \Sigma^+ \mid \sigma, 0 \models \varphi \}
\]

and for which the semantics of \( \text{LTL}(X, U) \) formulæ is adapted with the following, for any \( w = a_0a_1 \cdots a_n \in \Sigma^+ \) and index \( i \) in \( \mathbb{N} \):

\[
w, i \models X \varphi \quad \text{if } i < n \text{ and } w, i + 1 \models \varphi
\]

\[
w, i \models \psi U \varphi \quad \text{if } \exists k \text{ with } i \leq k \leq n, w, k \models \varphi \text{ and } \forall j \text{ with } i \leq j < k, w, j \models \psi
\]

Any aperiodic language \( L \subseteq \Sigma^\infty \setminus \{\varepsilon\} \) can be be also recognized by a morphism \( \mu : \Sigma^+ \rightarrow S \) into a finite aperiodic semigroup \( S \). This morphism induces two equivalence relations \( \sim_\mu \) on \( \Sigma^+ \) and \( \approx_\mu \) on \( \Sigma^\omega \), both of finite index, that saturate \( L (w \in L \text{ implies } [w] \subseteq L) \) and satisfy \([w] \cdot [v] \subseteq [u \cdot v]\) for \( u \) in \( \Sigma^+ \) and \( v \) in \( \Sigma^\infty \setminus \{\varepsilon\} \). Furthermore, each equivalence class is itself an aperiodic language.

**Exercise 4** (Separation into Past and Future). Let us define an \( \text{LTL}(Y, S) \) formula as pure past—it does not employ the \( X \) or \( U \) modalities. Conversely, an \( \text{LTL}(X, U) \) formula is pure future. The purpose of this exercise is to prove that any aperiodic language can be given a separation formulæ

\[
\varphi = \bigvee_{j \in J} \varphi_j \land a_j \land \varphi_j^3
\]

where \( J \) is some finite index set, and for each \( j \) in \( J \), \( a_j \) is a letter in \( \Sigma \) (or equivalently the formula \( \bigwedge_{p \in a_j} p \land \bigwedge_{p \in \text{AP} \setminus a_j} \neg p \), \( \varphi_j \) a pure past formula, and \( \varphi_j^3 \) a pure future formulæ.
1. Let $L \subseteq \Sigma^+$ be an aperiodic language of finite words. Show that $L$ can be associated with a pure past formula $\varphi$ such that

$$L = \{ w = a_0a_1 \cdots a_n \in \Sigma^+ \mid w, n \models \varphi \}.$$ 

2. Let $L \subseteq \Sigma^\omega$ be an aperiodic language. Prove that there exists a finite index set $J$ such that

$$L = \bigcup_{j \in J} P_j \cdot a_j \cdot F_j$$

with $a_j$ a letter in $\Sigma$, $P_j$ an aperiodic language included in $\Sigma^+$ or $\{\varepsilon\}$, and $F_j$ an aperiodic language included in $\Sigma^\omega$ for each $j$ of $J$.

3. Prove the following theorem (you can start by associating an LTL formula to each $P_j$ and each $F_j$):

**Theorem 2 (Separation).** Let $L \subseteq \Sigma^\omega$ be an aperiodic language. Then there exists a separation formula $\varphi = \bigvee_{j \in J} \overleftarrow{\varphi_j} \land a_j \land \overrightarrow{\varphi_j}$ such that

(i) $L = L(G\varphi)$,

(ii) $L = L(F\varphi)$,

(iii) $\text{Pref}(L) \setminus \{\varepsilon\} = \{ w = a_0a_1 \cdots a_n \in \Sigma^+ \mid w, n \models \bigvee_{j \in J} \overleftarrow{\varphi_j} \land a_j \}$, and

(iv) for each $j$ in $J$, the formula $\overleftarrow{\varphi_j} \land a_j \land \overrightarrow{\varphi_j}$ is satisfiable.

3 Characteristic LTL Formulae

This section characterizes LTL($Y, S, X, U$) formulæ $\varphi$ that describe safety or liveness properties.

**Exercise 5 (Characteristic Safety Formulae).** A characteristic safety formula is a formula of form $G\varphi$ where $\varphi$ is a pure past formula.

1. Provide a characteristic safety formula for the mutual exclusion property of Exercise 1.

2. Show that the language $L(G\varphi)$ of a characteristic safety formula is a safety property.

3. Let $\psi$ be an LTL($Y, S, X, U$) formula. Show that there exists a characteristic safety formula $G\varphi$ such that $\text{cl}(L(\psi)) = L(G\varphi)$.

**Exercise 6 (Characteristic Liveness Formulae).** A characteristic liveness formula is a formula of form $F \bigvee_{j \in J} (\overleftarrow{\varphi_j} \land a_j \land \overrightarrow{\varphi_j})$ for a finite index set $J$, where each $a_j$ is a letter from $\Sigma$, each $\overleftarrow{\varphi_j}$ a pure past formula, and each $\overrightarrow{\varphi_j}$ a pure future formula, such that $G(\bigvee_{j \in J} \overleftarrow{\varphi_j} \land a_j)$ is valid, and each $a_j \land \overrightarrow{\varphi_j}$ is a satisfiable formula.
1. Give a characteristic liveness formula for the starvation freedom property of Exercise 1.

2. Show that, if \( \varphi = F \bigvee_{j \in J} (\overline{e_j} \land a_j \land \overline{e_j}) \) is a characteristic liveness formula, then \( L(\varphi) \) is a liveness property.

3. Prove the converse, namely that if \( \psi \) is an LTL\((Y,S,X,U)\) formula such that \( L(\psi) \) is a liveness property, then there exists a characteristic liveness formula \( \varphi = F \bigvee_{j \in J} (\overline{e_j} \land a_j \land \overline{e_j}) \) such that \( L(\psi) = L(\varphi) \).

**Exercise 7** (Model Checking Safety Formulae). Given a pure past formula \( \varphi \) over a set of atomic propositions \( AP \), we want to construct a deterministic finite automaton \( A = (Q, \Sigma, T, q_0, F) \) over \( \Sigma = 2^{AP} \) that recognizes the language

\[
W_{\varphi} = \{ w = a_0 \cdots a_n \in \Sigma^+ \mid w, n \models \varphi \}.
\]

Let us define \( \text{sub}(\varphi) \) as the set of subformulæ of \( \varphi \), and set \( Q = 2^{\text{sub}(\varphi)} \).

1. Define a deterministic transition function \( T : Q \times \Sigma \to Q \) such that, for all \( w = a_0a_1 \cdots a_n \) of \( \Sigma^+ \),

\[
T(\emptyset, w) = \{ \psi \in \text{sub}(\varphi) \mid w, n \models \psi \}.
\]

Use it to show how to construct the desired automaton.

2. Show how to construct a deterministic Büchi automaton for a characteristic safety formula \( G\varphi \), such that all its states are accepting.

3. Show how to model check a system for a safety property expressed as a characteristic safety formula \( G\varphi \).

4. Prove that the model checking problem for finite Kripke structures and characteristic safety formulae is \( \text{PSPACE} \)-complete.
A Equivalence Relations Induced by $\mu$

For those interested by such matters, here is how the equivalence relations $\sim_\mu$ and $\approx_\mu$ can be defined: for all $u, v$ in $\Sigma^+$,

$$u \sim_\mu v \text{ iff } \mu(u) = \mu(v),$$

and for all $u, v$ in $\Sigma^\omega$,

$$u \sim_\mu v \text{ iff } \exists (u_i)_{i \in \mathbb{N}} \text{ and } (v_i)_{i \in \mathbb{N}}, u = u_0 u_1 u_2 \cdots, v = v_0 v_1 v_2 \cdots, \text{ and } \forall i \in \mathbb{N}, u_i \sim_\mu v_i,$$

from which one defines $\approx_\mu$ over $\Sigma^\omega$ as the transitive closure of $\sim_\mu$ over $\Sigma^\omega$. If needed, $\sim_\mu$ can be extended to $\Sigma^*$ by having $[\varepsilon] = \{\varepsilon\}$. 