Home Assignment 1: Safety and Liveness

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

October 2009

			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

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:

```
while (true)
  {
    /* 1: Noncritical section. */
    atomic { y_i = 1; s = i; };
    /* 2: Wait for turn. */
    wait until ((y_{1-i} == 0) || (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 Σ, the sets of finite words Σ^* and of infinite words Σ^ω, 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 Σ^ω 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 Σ^ω containing D is Σ^ω itself. We further define the set of finite prefixes of a language L included in Σ^ω as

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

and the *closure* of a language L included in Σ^ω as the language

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

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

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

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

Exercise 3 (Decomposition into Safety and Liveness)**.** Recall that a *property* over a set of atomic propositions AP is a language over $\Sigma = 2^{\mathrm{AP}}$, i.e. a subset of Σ^ω. 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 Σ^ω, and a *liveness* property as a dense subset of Σ^ω. 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 LTL($\mathsf{Y}, \mathsf{S}, \mathsf{X}, \mathsf{U}$) formula φ defines an *aperiodic language* $L(\varphi)$:

$$L(\varphi) = \{\sigma = a_0 a_1 a_2 \cdots \in \Sigma^\omega \mid \sigma, 0 \models \varphi\}$$

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

$$L(\varphi) = \{\sigma = a_0 a_1 \cdots a_n \in \Sigma^+ \mid \sigma, 0 \models \varphi\}$$

and for which the semantics of LTL(X, U) formulæ is adapted with the following, for any $w = a_0 a_1 \cdots a_n$ in Σ^+ and index i in $\mathbb{N}$:

$$\begin{aligned} &w, i \models \mathsf{X}\varphi && \text{if } i < n \text{ and } w, i+1 \models \varphi \\ &w, i \models \psi \,\mathsf{U}\, \varphi && \text{if } \exists k \text{ with } i \leq k \leq n, w, k \models \varphi \text{ and } \forall j \text{ with } i \leq j < k, w, j \models \psi \end{aligned}$$

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

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

$$\varphi = \bigvee_{j \in J} \overleftarrow{\varphi_j} \wedge a_j \wedge \overrightarrow{\varphi_j}$$

where J is some finite index set, and for each j in J, a_j is a letter in Σ (or equivalently the formula $\bigwedge_{p \in a_j} p \wedge \bigwedge_{p \in \mathrm{AP} \backslash a_j} \neg p$), $\overleftarrow{\varphi_j}$ a pure past formula, and $\overrightarrow{\varphi_j}$ a pure future formula.

1. Let $L \subseteq \Sigma^+$ be an aperiodic language of finite words. Show that L can be associated with a pure past formula φ such that
$$L = \{w = a_0 a_1 \cdots a_n \in \Sigma^+ \mid w, n \models \varphi\} .$$

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 Σ, P_j an aperiodic language included in Σ^+ or $\{\varepsilon\}$, and F_j an aperiodic language included in Σ^ω 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 $\varphi = \bigvee_{j \in J} \overleftarrow{\varphi_j} \wedge a_j \wedge \overrightarrow{\varphi_j}$ such that*

(i) $L = L(\mathsf{G}\varphi)$,

(ii) $L = L(\mathsf{F}\varphi)$,

(iii) $\mathsf{Pref}(L) \backslash \{\varepsilon\} = \{w = a_0 a_1 \cdots a_n \in \Sigma^+ \mid w, n \models \bigvee_{j \in J} \overleftarrow{\varphi_j} \wedge a_j\}$, *and*

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

3 Characteristic LTL Formulæ

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

Exercise 5 (Characteristic Safety Formulæ)**.** A *characteristic safety formula* is a formula of form $\mathsf{G}\varphi$ where φ is a pure past formula.

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

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

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

Exercise 6 (Characteristic Liveness Formulæ)**.** A *characteristic liveness formula* is a formula of form $\mathsf{F} \bigvee_{j \in J} (\overleftarrow{\varphi_j} \wedge a_j \wedge \overrightarrow{\varphi_j})$ for a finite index set J, where each a_j is a letter from Σ, each $\overleftarrow{\varphi_j}$ a pure past formula, and each $\overrightarrow{\varphi_j}$ a pure future formula, such that $\mathsf{G}(\bigvee_{j \in J} \overleftarrow{\varphi_j} \wedge a_j)$ is valid, and each $a_j \wedge \overrightarrow{\varphi_j}$ is a satisfiable formula.

1. Give a characteristic liveness formula for the starvation freedom property of Exercise 1.

2. Show that, if $\varphi = \mathsf{F} \bigvee_{j \in J} (\overleftarrow{\varphi_j} \wedge a_j \wedge \overrightarrow{\varphi_j})$ is a characteristic liveness formula, then $L(\varphi)$ is a liveness property.

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

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

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

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

1. Define a deterministic transition function $T : Q \times \Sigma \to Q$ such that, for all $w = a_0 a_1 \cdots a_n$ of Σ^+,

   $$T(\emptyset, w) = \{\psi \in \mathsf{sub}(\varphi) \mid w, n \models \psi\} \ .$$

   Use it to show how to construct the desired automaton.

2. Show how to construct a deterministic Büchi automaton for a characteristic safety formula $\mathsf{G}\varphi$, 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 $\mathsf{G}\varphi$.

4. Prove that the model checking problem for finite Kripke structures and characteristic safety formulæ is PSPACE-complete.

A Equivalence Relations Induced by μ

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 Σ^+,

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

and for all u, v in Σ^ω,

$$u \sim_\mu v \quad \text{iff } \exists (u_i)_{i\in\mathbb{N}} \text{ and } (v_i)_{i\in\mathbb{N}}, u = u_0u_1u_2\cdots, v = v_0v_1v_2\cdots, \text{ and } \forall i \in \mathbb{N}, u_i \sim_\mu v_i,$$

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