TD 1

1 Models

Exercise 1 (Rendez-vous with Data). Consider the synchronization of transition systems with variables through a rendez-vous mechanism. Such a system is of form $M = (S, \Sigma, \mathcal{V}, (D_v)_{v \in \mathcal{V}}, T, I, AP, l)$ where $\mathcal{V}$ the set of (typed) variables $v$, each with domain $D_v$.

We want to extend the rendez-vous mechanism between systems with variables with the ability to exchange data values. For instance, a system $M_i$ may transmit a value $m$ with guard $g$ and label $a$ by performing

$$s_i \xrightarrow{!m} s'_i ,$$

only if some system $M_j$ is ready to receive the message, i.e. to perform

$$s_j \xrightarrow{?v} s'_j ,$$

where $v$ is a variable of $M_j$ and $m$ is in $D_v$. Of course the synchronization is also possible if $M_j$ performs instead

$$s_j \xrightarrow{?m} s'_j .$$

Propose Structural Operational Semantics for the rendez-vous with data synchronization.

Exercise 2 (Needham-Schroeder Protocol). We consider the analysis of a public-key authentication protocol proposed by Needham and Schroeder in 1978. The protocol relies on

- the generation of *nounces* $NC$ : random numbers that should only be used in a single session, and
- on public key encryption : we denote the encryption of message $M$ using $C$’s public key by $\langle M \rangle_C$.

A(lice) and B(ob) try to make sure of each other’s identity by the following (very simplified) exchange :

1.
1. Alice first presents herself (the $A$ part of the message) and challenges Bob with her nounce $N_A$. Assuming both cryptography and random number generation to be perfect, only Bob can decrypt $\langle A, N_A \rangle_B$ and find the correct number $N_A$.

2. Bob responds by proving his identity (the $N_A$ part) and challenges Alice with his own nounce $N_B$.

3. Finally, Alice proves her identity by sending $N_B$.

The nounces $N_A$ and $N_B$ are used by Alice and Bob as secret keys for their communications.

In order to account for the insecure channel, we have to add an intruder $I$ to the model, who has his own nounce $N_I$, and can read and send any message it fancies, but can only decrypt $\langle M \rangle_I$ messages and cannot guess the nounces generated by Alice and Bob.

We can model the behaviour of Alice as a transition system $M_A$ with variables and rendez-vous with data, using a single variable $N$ ranging over $\mathcal{N} = \{N_A, N_B, N_I\}$.

1. Provide a model $M_B$ for Bob.
2. Provide a model $M_I$ the intruder.
3. Give a LTL formula that states that the intruder cannot know $N_A$ and $N_B$, nor make Bob believe he is Alice.
4. Unfold an execution path in the synchronized product of $M_A$, $M_B$, and $M_I$ that unveils a flaw in the protocol.
2 Specification

Exercise 3 (LTL Formulæ). We would like to verify the properties of a boolean circuit with input $x$, output $y$, and two registers $r_1$ and $r_2$. We define accordingly $\text{AP} = \{x, y, r_1, r_2\}$ as our set of atomic propositions, and model check infinite runs $\sigma = s_0 s_1 s_2 \cdots$ from $(2^{\text{AP}})^\omega$.

Translate the following properties in LTL and in FO(<):
1. “it is impossible to get two consecutive 1 as output”
2. “each time the input is 1, at most two ticks later the output will be 1”
3. “each time the input is 1, the register contents remains the same over the next tick”
4. “register $r_1$ is infinitely often 1”

3 Automata

Definition 1 (Büchi Automaton). A Büchi automaton is a tuple $A = (Q, \Sigma, T, I, F)$ where $Q$ is a finite set of states, $\Sigma$ a finite alphabet, $T \subseteq Q \times \Sigma \times Q$ a transition relation, $I \subseteq Q$ a set of initial states, and $F \subseteq Q$ a set of accepting states.

An infinite run $\sigma = q_0 q_1 q_2 \cdots$ in $Q^\omega$ for an infinite word $w = a_0 a_1 a_2 \cdots$ in $\Sigma^\omega$ is successful if $q_0$ is in $I$, for each $i$ $(q_i, a_i, q_{i+1})$ is in $T$, and $\sigma$ visits $F$ infinitely often, written $\text{Inf}(\sigma) \cap F \neq \emptyset$. The language $L(A)$ of $A$ is the set of words that have at least one successful run. We say that a language $L \subseteq \Sigma^\omega$ is recognizable, noted $L \in \text{Rec}(\Sigma^\omega)$, if it is the language of some Büchi automaton.

Exercise 4 (Generalized Acceptance Condition). A generalized Büchi automaton $A = (Q, \Sigma, T, I, (F_i)_i)$ has a finite set of accepting sets $F_i$. An infinite run is satisfies this generalized acceptance condition if

$$\bigwedge_i \text{Inf}(\sigma) \cap F_i \neq \emptyset.$$

Show that for any generalized Büchi automaton, one can construct an equivalent (non generalized) Büchi automaton.

Exercise 5 (Basic Closure Properties). Show that $\text{Rec}(\Sigma^\omega)$ is closed under
1. finite union, and
2. finite intersection.
**Exercise 6** (Ultimately Periodic Words). An *ultimately periodic word* over $\Sigma$ is a word of form $u \cdot v^\omega$ with $u$ in $\Sigma^*$ and $v$ in $\Sigma^+$. 

Prove that any nonempty recognizable language in $\text{Rec}(\Sigma^\omega)$ contains an ultimately periodic word.

**Exercise 7** (Rational Languages). A *rational language* $L$ of infinite words over $\Sigma$ is a finite union 

$$L = \bigcup X \cdot Y^\omega$$

where $X$ is in $\text{Rat}(\Sigma^*)$ and $Y$ in $\text{Rat}(\Sigma^+)$. We denote the set of rational languages of infinite words by $\text{Rat}(\Sigma^\omega)$.

Show that $\text{Rec}(\Sigma^\omega) = \text{Rat}(\Sigma^\omega)$.

**Exercise 8** (Deterministic Büchi Automata). A Büchi automaton is *deterministic* if $|I| \leq 1$, and for each state $q$ in $Q$ and symbol $a$ in $\Sigma$, $|\{(q, a, q') \in T \mid q' \in Q\}| \leq 1$.

1. Give a nondeterministic Büchi automaton for the language in $\{a, b\}^\omega$ described by the expression $(a + b)^* a^\omega$.
2. Show that there does not exist any deterministic Büchi automaton for this language.
3. Let $A = (Q, \Sigma, T, q_0, F)$ be a finite deterministic automaton that recognizes the language of finite words $L \subseteq \Sigma^*$. What is the language of infinite words recognized by $A$ seen as a Büchi automaton?