Disclaimer: Rough outline of the solutions! You can let me know if you see typos.

Fix a set of atoms $Prop$ and labels $\mathcal{L}$. Given two transition systems $\mathcal{T} = (S, \to, \rho)$ and $\mathcal{T}' = (S', \to', \rho')$ using those parameters (i.e., $\to \subseteq S \times \mathcal{L} \times S$, $\rho : Prop \to \mathcal{P}(S), \dots$), a *bisimulation* between $\mathcal{T}$ and $\mathcal{T}'$ is a relation $\sim \subseteq S \times S'$ such that, for every $s \sim s'$ we have the conjunction of the following:

- for every $P \in Prop$, $s \in \rho(P) \Leftrightarrow s' \in \rho'(P)$ (bisimilar states satisfy the same atomic propositions)
- for every transition $s \xrightarrow{\tau} r$ (for every $r \in S$ and $\tau \in \mathcal{L}$), there exists $r' \in S'$ with $r \sim r'$ and a transition $s' \xrightarrow{\tau} r'$
- for every transition $s' \xrightarrow{\tau} r'$ (for every $r' \in S'$ and $\tau \in \mathcal{L}$), there exists $r \in S$ with $r \sim r'$ and a transition $s \xrightarrow{\tau} r$.

Given two arbitrary LTS, there always exists a *maximal bisimulation* which can be defined as the union of all bisimulations. We say that two states s and s' in two transition systems (not necessarily distinct!) are bisimilar (and write it simply $s \sim s'$ from now on) if they sit in any bisimulation, or equivalently, in the maximal bisimulation.

Recall that in a previous exercise, we showed that if $s \sim s'$, then for every *closed*[1] μ-calculus formula φ, $s \in ||\varphi||^{\mathcal{T}} \Leftrightarrow s' \in ||\varphi||^{\mathcal{T}'}$.

Question 1. *For the purpose of this question, call a transition system* based on a tree *if its state space is a prefix-closed subset of Σ^* for some set Σ (possibly infinite!) and such that its transition relation $\to$ satisfies the following: for every $u, v \in \Sigma^*$ and $\tau \in \mathcal{L}$, if $u \xrightarrow{\tau} v$, then $v = ua$ for some $a \in \Sigma$.*

Show that for every LTS $\mathcal{T} = (S, \to, \rho)$ and state $s_0 \in S$, there is another LTS based on a tree such that its root is bisimilar to s_0.

Solution. Consider the LTS $\mathcal{U} = (S_{\mathcal{U}}, \to_{\mathcal{U}}, \rho_{\mathcal{U}})$ defined as follows:

- $S_{\mathcal{U}}$ is the set of words in $(\mathcal{L} \times S)^*$ which describe a valid path starting from s_0 in $\mathcal{T}$. In symbols:

  $(\tau_1, s_1)(\tau_2, s_2) \dots (\tau_k, s_k) \in S_{\mathcal{U}}$ iff $s_0 \xrightarrow{\tau_1} s_1 \xrightarrow{\tau_2} s_2 \dots \xrightarrow{\tau_k} s_k$ is a path in $\mathcal{T}$

  By convention, the empty word is also in $S_{\mathcal{U}}$ and corresponds to the empty path. Using the function $last : (\mathcal{L} \times S)^* \to S$ mapping the empty word to s_0 and a non-empty word $u(\tau, s)$ to s, this can be more pedantically characterized as the least subset of $(\mathcal{L} \times S)^*$

  - containing the empty word
  - such that, for every $u \in S_{\mathcal{U}}$ and transition $last(u) \xrightarrow{\tau} s$, we also have $u(\tau, s) \in S_{\mathcal{U}}$.

[1] This can be extended to a result for open formulas in the way I tried to address in class 3.

- $\rightarrow_{\mathcal{U}}$ is the least relation such that $u \xrightarrow{\tau} u(\tau, s)$ for every transition $last(u) \xrightarrow{\tau} s$ in $\mathcal{T}$.
- the valuation of atomic predicate $\rho_{\mathcal{U}}$ is equal to $\rho \circ last$.

With this definition, one can check that the relation

$$\{(last(u), u) \mid u \in S_{\mathcal{U}}\} \quad \subseteq \quad S \times S_{\mathcal{U}}$$

is *a* bisimulation[2] and that we indeed have $s_0 \sim \varepsilon$. (left to the reader) □

For simplicity, I shall only consider LTS where $\mathcal{L}$ is a singleton $\{\bullet\}$ from now on. I will write $\Box/\Diamond$ instead of $[\bullet]/\langle\bullet\rangle$ and abuse notations to suppose that $\rightarrow \subseteq S \times S$ in such LTSs $(S, \rightarrow, \rho)$.

Question 2. *Consider the following property, meant to be evaluated at some initial node s_0 of some LTS. Are they expressible in the μ-calculus?*

1. *There exists a unique infinite path on which the atomic proposition P is always true.*
2. *On every branch, if P holds twice, then Q must have held once (strictly) in between.*
3. *There exists a loop reachable from s_0.*
4. *P is true at s iff for every path from s_0 to s, Q is holds on an even number of steps (strictly before reaching s).*

Solution. 1. No. Consider the LTS $\mathcal{T}_\omega = (\mathbb{N}, \{(n, n+1) \mid n \in \mathbb{N}\}, P \mapsto \mathbb{N})$ and $(\mathbb{N}, \{(n, n+2) \mid n \in \mathbb{N}\} \cup \{(0, 1)\}, P \mapsto \mathbb{N})$. The states 0 are clearly bisimilar, but the first LTS has a unique infinite P-path and the second has two of them. Therefore this property cannot be captured by a μ-calculus formula using the exercise from last week.

2. Yes. Consider

$$\nu X.\Box X \wedge P \Rightarrow \Box(\mu Y.\neg P \wedge (\Box Y \vee Q \vee (\nu Z.\Box Z \wedge \neg P)))$$

The outer ν ensures that the property holds everywhere, the inner μ enforces that whenever P is encountered, we have necessarily $\neg P$ until either Q holds or there is no possibilities of encountering a P ever ($\nu Z.\Box Z \wedge \neg P$).

3. No; for instance, the LTS $\mathcal{T}_\omega$ and $(\{\bullet\}, \{(\bullet, \bullet)\}, P \mapsto \{\bullet\})$ have their initial state which are bisimilar, but there is no loop in $\mathcal{T}_\omega$ while there is one in the second.

[2] It is not necessarily maximal

4. Yes. Consider

$$\nu E.P \wedge (Q \Rightarrow \Box(\nu O.\neg P \wedge (Q \Rightarrow \Box E) \wedge (\neg Q \Rightarrow \Box O))) \wedge (\neg Q \Rightarrow \Box E)$$

This was meant as a warm-up for the next exercise. To ground intuitions, O should be read as "Odd" and E as "Even".

□

From now on, I'm going to try to relate material on S1S, LTL and μ-calculus. To do so, I will restrict my attention to LTSs that are generated by infinite words on a fixed alaphabet $\mathcal{P}(Prop)$. Given $u \in \mathcal{P}(Prop)^\omega$, define the LTS $\mathcal{T}_u$ as $(\mathbb{N}, \{(n, n+1) \mid n \in \mathbb{N}, P \mapsto \{n \mid P \in u_n\})$. For such transition system, note that there is no difference between $\Box$ and $\Diamond$. I will simply write $\bigcirc$ instead, which you can read as "next".

Question 3. *Call a triple $\mathcal{A} = (Q, q_0, \delta)$ where a Q is a finite set, $q_0 \in Q$ and δ is a* partial function $\mathcal{P}(Prop) \times Q \rightharpoonup Q$ *a deterministic*[3] *safety automaton for the purpose of this exercise. Say that $u \in \mathcal{P}(Prop)^\omega$ is recognized by $\mathcal{A}$ if there exists a (unique) run $\rho \in Q^\omega$ of $\mathcal{A}$ over u (it is not necessarily the case since δ is partial).*

Describe a procedure turning any deterministic safety automaton into a μ-calculus formula $\varphi_\mathcal{A}$ such that $0 \in ||\varphi_\mathcal{A}||^{\mathcal{T}_u}$ iff u is recognized by $\mathcal{A}$.

Proof idea. Suppose that $Q = \{q_0, q_1, \ldots, q_n\}$. To each state $q_i \in Q$, associate a set variable X_i. For every $u \in \mathcal{P}(Prop)^\omega$, the valuation

$$V : X_i \mapsto \{n \mid \text{there exists a run of } (Q, q_i, \delta) \text{ over the suffix } u_n u_{n+1} \ldots\}$$

is the greatest (for pointwise inclusion, in the complete lattice $\mathcal{P}(Prop) \to \mathcal{P}(\mathbb{N})$) such that the following holds at every node

$$X_i \text{ iff } \bigwedge_{T \subseteq Prop} \left[\left(\bigwedge_{P \in T} P \wedge \bigwedge_{P \notin T} \neg P \right) \Rightarrow \bigcirc \begin{cases} X_{\delta(q_i,T)} & \text{when defined} \\ \bot & \text{otherwise} \end{cases} \right]$$

Call φ_i the formula on the right-hand side of this equivalence. For every $i \leq k \leq n$, we define a μ-calculus formula $\psi_i^{(k)}$ with free variables X_j for $j < i$ such that $n \in ||\psi_i^{(k)}||_V^{\mathcal{T}_u}$ if and only if there exists a run of (Q, q_i, δ) over $u_n u_{n+1} \ldots$. This can be done by by setting, for $k = i$

$$\psi_i^{(i)} \quad = \quad \nu X_i . \psi_i^{(i+1)} [\psi_{i+1}^{(i+1)} / X_{i+1}, \ldots, \psi_n^{(i+1)} / X_n]$$

and, for $i < k$,

$$\psi_i^{(k)} \quad = \quad \psi_i [\psi_k^{(k)} / X_k, \ldots, \psi_n^{(k)} / X_n]$$

We then have a closed μ-calculus formula $\psi_0^{(0)}$ which holds on the initial node of $\mathcal{T}_u$ if and only if there exists an infinite run of $\mathcal{A}$. □

[3] Not essential; for our purpose, these things can be determinized by a powerset construction anyway.

Question 4. *Give a translation $\varphi \to \varphi^*$ from LTL formulas to μ-calculus such that*

$$u \models \varphi \qquad \textit{iff} \qquad 0 \in ||\varphi^*||^{\mathcal{T}_u}$$

Solution. φ^* is computed inductively over the structure of φ (I will limit myself to a minimal set of connectives for LTL). The proof of soundness, which is doable by structural induction over φ, is left to the reader.

$$\begin{array}{lcllcl} P^* & = & P & (\varphi \wedge \psi)^* & = & (\varphi \wedge \psi)^* \\ (\neg\varphi)^* & = & \neg\varphi^* & (G\varphi)^* & = & \nu Y.\varphi^* \wedge \bigcirc Y \\ (X\varphi)^* & = & \bigcirc\varphi^* & (\varphi U \psi)^* & = & \mu Y.\psi^* \vee (\varphi^* \wedge \bigcirc Y) \end{array}$$

□

Question 5. *Similarly, give a translation from μ-calculus to S1S.*

Solution. Here we will need to take as input μ-calculus formulas with whose set of free propositional variables may vary inductively, so we need to be careful about the invariant we want our translation to maintain. A μ-calculus formula $\varphi(P_1, \ldots, P_k)$ with the displayed propositional variables will be interpreted as a S1S formula $\varphi(P_1, \ldots, P_k, n)^\dagger$ using the same variables as set/predicate variable, plus an extra individual variable n standing for a position in the infinite word u.

At the semantic level, we are going to maintain the following invariant, for every $u \in \{P_1, \ldots, P_k\}^\omega$ and concrete natural number n:

$$u \models \varphi(P_1, \ldots, P_k, n)^\dagger \qquad \text{iff} \qquad n \in ||\varphi(P_1, \ldots, P_k)||^{\mathcal{T}_u}$$

Once again, we define the translation $\varphi \mapsto \varphi^\dagger$ by induction over the size of φ and I leave to the reader the task of checking that this translation satisfies the above soundness property. First, much like previously, this translation commutes with propositional logical connectives

$$(\varphi \wedge \psi)^\dagger \quad = \quad \varphi^\dagger \wedge \psi^\dagger \qquad (\varphi \vee \psi)^\dagger \quad = \quad \varphi^\dagger \vee \psi^\dagger$$

For atoms P_i, we set $P_i^\dagger = n \in P_i$. We also use our extra free variable when encountering a modality:

$$(\bigcirc\varphi(P_1, \ldots, P_k, n))^\dagger \quad = \quad \varphi(P_1, \ldots, P_k, n+1)$$

Finally for the fixpoint operators, one can take a leaf from the proof of the Knaster-Tarski theorem. Let's deal with a the translation of a least fixpoint formula $\mu P_{k+1}.\varphi(P_1, \ldots, P_k, P_{k+1})$. If we represent the set of position satisfying this formula as a set X, we know that it is a prefixpoint of φ. This can be translated in μ-calculus as $\varphi(P_1, \ldots, P_k, X) \Rightarrow X$; translated to S1S, it means that we should have

$$\varphi(P_1, \ldots, P_k, X, n)^\dagger \Rightarrow n \in X$$

Moreover, we know that the actual least fixpoint is the intersection of every prefixpoint; we can capture this in S1S using universal quantification

$$m \in X \quad \Leftrightarrow \quad \forall Z.(\forall n \in \mathbb{N}(\varphi(P_1, \ldots, P_k, Z, n)^\dagger \Rightarrow n \in Z)) \Rightarrow m \in Z$$

As a result, we can safely set

$$(\mu P_{k+1}.\varphi(P_1, \ldots, P_k, P_{k+1}, m))^\dagger \;=\; \forall Z.(\forall n \in \mathbb{N}(\varphi(P_1, \ldots, P_k, Z, n)^\dagger \Rightarrow n \in Z)) \Rightarrow m \in Z$$

Dually, we can take

$$(\nu P_{k+1}.\varphi(P_1, \ldots, P_k, P_{k+1}, m))^\dagger \;=\; \exists Z.(\forall n \in \mathbb{N}(n \in Z \Rightarrow \varphi(P_1, \ldots, P_k, Z, n)^\dagger) \wedge m \in Z$$

You may have seen this sort of definition elsewhere called "impredicative encodings of (co)-inductive definitions". These crucially use second-order quantification in S1S. □

Last comments:

- There is a converse translation from S1S infinite word automata to μ-calculus; this is however more involved!
- LTL and S1S only deal with infinite words; there are variants of these logics dealing with branching related to the modal μ-calculus as you have seen in the lectures. If you have seen CTL/CTL* in other courses, you can think about translating them to μ-calculus (much like with LTL in Question 4).