Automata and Logic on Infinite Objects 2

Shay Kricheli

December 2019

Question 1

To answer this question, let us remind:

Definition 0.1. *A language $L \subseteq \Sigma^w$ is non-counting if and only if:*

$$\exists n_0 \in \mathbb{N} \text{ s.t. } \forall n \geq n_0 \ ; \ \forall u \in \Sigma^*, v \in \Sigma^+, w \in \Sigma^w \ ; \ uv^n w \in L \iff uv^{n+1} w \in L$$

In this question, we are to complete the proof for the following:

Claim 0.2. *For every LTL formula φ, the set $[\![\varphi]\!]$ is non-counting.*

The proof is by structural induction on φ and we are to complete the case where $\varphi = \varphi_1 U \varphi_2$. Let $\varphi = \varphi_1 U \varphi_2$. Let us denote $L_1 = [\![\varphi_1]\!]$ and $L_2 = [\![\varphi_2]\!]$. By the induction hypothesis, we have that L_1 and L_2 are non-counting. Let n_1, n_2 be the constants promised by the induction hypothesis for L_1 and L_2 respectively such that:

$$\forall n \geq n_1 \ ; \ \forall u \in \Sigma^*, v \in \Sigma^+, w \in \Sigma^w \ ; \ uv^n w \in L_1 \iff uv^{n+1} w \in L_1$$
$$\forall n \geq n_2 \ ; \ \forall u \in \Sigma^*, v \in \Sigma^+, w \in \Sigma^w \ ; \ uv^n w \in L_2 \iff uv^{n+1} w \in L_2$$

By the definition of a model of a LTL formula, these correspond to:

$$\forall n \geq n_1 \ ; \ \forall u \in \Sigma^*, v \in \Sigma^+, w \in \Sigma^w \ ; \ uv^n w \models \varphi_1 \iff uv^{n+1} w \models \varphi_1$$
$$\forall n \geq n_2 \ ; \ \forall u \in \Sigma^*, v \in \Sigma^+, w \in \Sigma^w \ ; \ uv^n w \models \varphi_2 \iff uv^{n+1} w \models \varphi_2$$

Let us choose $n_0 = \max\{n_1, n_2\} + 1$ and let $n \in \mathbb{N}$ such that $n > n_0$. We saw in class that:

$$\forall u \in \Sigma^*, v \in \Sigma^+, w \in \Sigma^w \ ; \ \ uv^n w \models \varphi_1 U \varphi_2 \Longrightarrow uv^{n+1} w \models \varphi_1 U \varphi_2$$

We are to show that:

$$\forall u \in \Sigma^*, v \in \Sigma^+, w \in \Sigma^w \ ; \ \ uv^{n+1} w \models \varphi_1 U \varphi_2 \Longrightarrow uv^n w \models \varphi_1 U \varphi_2$$

Let $u \in \Sigma^*, v \in \Sigma^+, w \in \Sigma^w$ and let us assume that $uv^{n+1} w \models \varphi_1 U \varphi_2$. By the definition of the "until" operator - this implies:

$$\exists k \text{ s.t. } uv^{n+1} w[k..] \models \varphi_2 \wedge \forall j < k \ ; \ uv^{n+1} w[j..] \models \varphi_1$$

Let k be the least such k. Let us now consider two cases:

- $k < |u| + |v|$

  In that case we have:

$$uv^{n+1}w[k..] = uv[k..]v^n w \wedge \forall j < k\ ;\ uv^{n+1}w[j..] = uv[j..]v^n w$$

  Let $u' = uv$ and $n' = n - 1$. We know that: $n > max(n_1, n_2) + 1$ and therefore $n' = n - 1 > max(n_1, n_2)$ and thus the induction hypothesis applies to n'. Thus:

$$uv^{n+1}w[k..] = uv[k..]v^n w = u'[k..]v^{n'+1}w = u'v^{n'+1}w[k..] \wedge$$
$$\forall j < k\ ;\ uv^{n+1}w[j..] = uv[j..]^n w = u'[j..]v^{n'+1}w = u'v^{n'+1}w[j..]$$
$$\rightarrow u'v^{n'+1}w[k..] \models \varphi_2 \wedge \forall j < k\ ;\ u'v^{n'+1}w[j..] \models \varphi_1$$

  and by the induction hypothesis we have:

$$u'v^{n'}w[k..] \models \varphi_2 \wedge \forall j < k\ ;\ u'v^{n'}w[j..] \models \varphi_1$$

  which is equivalent to:

$$uvv^{n-1}w[k..] = uv^n w[k..] \models \varphi_2 \wedge \forall j < k\ ;\ uvv^{n-1}w[j..] = uv^n w[j..] \models \varphi_1$$

  and thus $uv^n w \models \varphi_1 U \varphi_2$.

- $k \geq |u| + |v|$

  In that case, let us consider $uv^{n+1}w[i..]$ It may be that it equals $u'v^m w[i..]$ for $m < n_0$, so the induction hypothesis does not apply in that case. However, $uv^{n+1}w[k..] \models \varphi_2$ implies $uv^n w[k - |v|..] \models \varphi_2$ since they agree on the inspected suffix. From the same arguments, $\forall (|u| + |v|) \leq j < k\ ;\ uv^n w[j - |v|..] \models \varphi_1$.

  For $j < (|u| + |v|)$ from the same explanation presented above (case $k < |u| + |v|$), we get that $\forall j < (|u| + |v|)\ ;\ uv^n w[j..] \models \varphi_1$

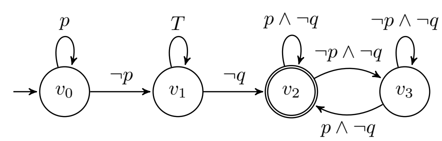
  Therefore, $uv^{n+1}w \models \varphi_1 U \varphi_2 \Longrightarrow uv^n w \models \varphi_1 U \varphi_2$

Question 2

We saw in class that $\mathbb{LTL} \subseteq \mathbb{NBW}$, thus, for each of the formulas ψ_i, we will present an NBW $\mathcal{B}_i$ over $\Sigma = 2^{AP}$, such that $[\![\mathcal{B}_i]\!] = [\![\psi_i]\!]$. Let us consider the following:
$\varphi_0 = Gp \vee Gq$
$\neg\varphi_0 = \neg(Gp \vee Gq) = \neg Gp \wedge \neg Gq = F\neg p \wedge F\neg q$
$\varphi_1 = GFp$
$\neg\varphi_1 = \neg GFp = F\neg(Fp) = FG\neg p$
$\varphi_2 = GFp \to GFq$
$\neg\varphi_2 = \neg(GFp \to GFq) = GFp \wedge \neg GFq = GFp \wedge F\neg(Fq) = GFp \wedge FG\neg q$
$\varphi_3 = FGp \wedge GFp$
Let us construct the automatons:

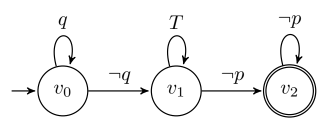
$$\psi_1 = \neg\varphi_0 \wedge \varphi_1\neg\varphi_2 = (F\neg p \wedge F\neg q) \wedge (GFp) \wedge (GFp \wedge FG\neg q) = F\neg q \wedge GFp \wedge FG\neg q$$

$\mathcal{B}_1$:



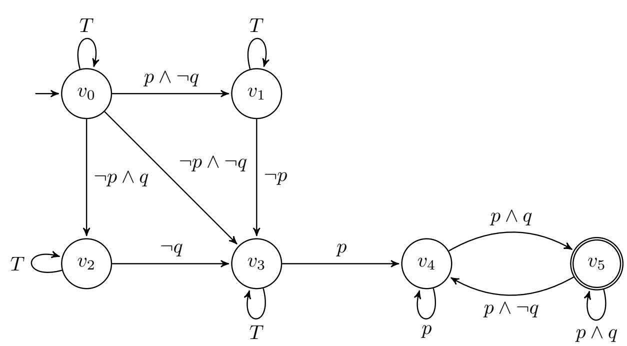
$$\psi_2 = \neg\varphi_0 \wedge \neg\varphi_1 \wedge \varphi_2 = (F\neg p \wedge F\neg q) \wedge (FG\neg p) \wedge (GFp \to GFq) = F\neg q \wedge FG\neg p$$

$\mathcal{B}_2$:



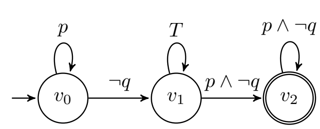
$$\psi_3 = \neg\varphi_0 \wedge \varphi_2 \wedge \varphi_3 = (F\neg p \wedge F\neg q) \wedge (GFp \to GFq) \wedge (FGp \wedge GFp) = F\neg p \wedge F\neg q \wedge FGp \wedge GFq$$

$\mathcal{B}_3$:



$$\psi_4 = \neg\varphi_0 \wedge \neg\varphi_2 \wedge \varphi_3 = (F\neg p \wedge F\neg q) \wedge (GFp \wedge FG\neg q) \wedge (FGp \wedge GFp) = F\neg p \wedge F\neg q \wedge GFp \wedge FG\neg q \wedge FGp = F\neg p \wedge FGp \wedge FG\neg q$$

$\mathcal{B}_4$:



Question 3

(1)

Let us consider the following restricted grammer:

$$\varphi :: r \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid r \mapsto \varphi$$

where r is a regular expression. Let us denote:

$$r \Mapsto \varphi ::= r \mapsto (true, \varphi)$$
$$\varphi_1 \wedge \varphi_2 ::= \neg(\neg\varphi_1 \wedge \neg\varphi_2)$$

In this question we are to describe the following languages using PSL formulas in that restricted grammar:

- p never holds:

$$(true)^+ \mapsto \neg p$$

- p holds on every third cycle, starting from an even position:

$$(true \cdot true)^* \cdot (true \cdot p) \Mapsto G(true \cdot true \cdot p)$$

- p_1 holds on every third cycle in which p_2 holds:

$$\Big((\neg p_2)^* \cdot p_2 \cdot (\neg p_2)^* \cdot p_2 \cdot (\neg p_2)^* \cdot p_2\Big)^+ \mapsto p_1$$

- p_1 holds forever long starting the cycle where p_2 held for 3 consecutive cycles:

$$\Bigg(\Big(\neg\Big(true^* \cdot p_2 \cdot p_2 \cdot p_2 \cdot true^*\Big)\Big) \cdot (p_2 \wedge p_1)^3 \cdot true^* \mapsto p_1\Bigg) \wedge$$
$$\Bigg(\Big(true^* \cdot \big((p_2 \wedge \neg p_1) \cdot p_2 \cdot p_2 + p_2 \cdot (p_2 \wedge \neg p_1) \cdot p_2 + p_2 \cdot p_2 \cdot (p_2 \wedge \neg p_1)\Big) \mapsto false\Bigg)$$

(2)

Let us consider the following PSL formulas:

$$\varphi_1 = (p_1 \wedge Xp_2)\ U\ (p_3 \wedge Xp_4)$$
$$\varphi_2 = (p_3 \cdot p_4) \vee \Big((p_1 \cdot p_2) \wedge (p_1 \cdot p_2)^+ \Mapsto (p_3 \cdot p_4)\Big)$$

In this section we are to prove or give a counterexample for the claim: "the following two PSL formulas are equivalent". Let us provide a counterexample and let us assume a word is formulated using 4-dimensional vectors such that the i'th coordinate corresponds to p_i for every $i \in [1, 4]$. Let:

$$w = \begin{bmatrix}1\\0\\0\\0\end{bmatrix} \begin{bmatrix}0\\1\\1\\0\end{bmatrix} \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

Claim 0.3. $w \models_{PSL} \varphi_1$

Proof. Since $p_3 \in w[2]$ and $p_4 \in w[3]$ we have that for $k = 2$, $w[k..] \models_{\text{PSL}} (p_3 \wedge Xp_4)$. Moreover, since $p_1 \in w[1]$ and $p_2 \in w[2]$ we have that $w[1..] \models_{\text{PSL}} (p_1 \wedge Xp_2)$. Since $j = 1$ is the only index that holds $j < k$, by the definition of the "until" operator, we have that $w \models_{\text{PSL}} (p_1 \wedge Xp_2)\, U\, (p_3 \wedge Xp_4) = \varphi_1$. □

Claim 0.4. $w \not\models_{PSL} \varphi_2$

Proof. Since $p_3 \notin w[1]$ we have that $w \not\models_{\text{PSL}} (p_3 \cdot p_4)$. Since $p_1 \in w[1]$ and $p_2 \in w[2]$ we have that for $k = 2$, $w[..k] \in [\![p_1 \cdot p_2]\!]$ and thus $w \models_{\text{PSL}} (p_1 \cdot p_2)$, but since $p_3 \notin w[3]$ we have that $w[k+1..] \not\models_{\text{PSL}} (p_3 \cdot p_4)$ and thus $w \not\models_{\text{PSL}} \Big((p_1 \cdot p_2) \wedge (p_1 \cdot p_2)^+ \mapsto (p_3 \cdot p_4)\Big)$. Thus $w \not\models_{\text{PSL}} (p_3 \cdot p_4) \vee \Big((p_1 \cdot p_2) \wedge (p_1 \cdot p_2)^+ \mapsto (p_3 \cdot p_4)\Big) = \varphi_2$. □

Thus we have that φ_1 and φ_2 are not equivalent.

Question 4

In this question we are asked to decide for each of the given languages over $\Sigma = 2^{\{p,q\}}$ if they can be accepted by an LTL formula and by a PSL formula. A point to notice is that LTL syntax is subsumed by PSL syntax and therefore wherever we have an LTL formula for a language, it's also the corresponding PSL formula.

1. $L_1 = \{w : p \in w[i] \wedge q \notin w[i]\ \forall i \geq 3\}$
   Let us define the following formulas in their corresponding logic:
   (i) LTL - $X^2p \wedge X^2G(\neg q)$
   (ii) PSL - $X^2p \wedge X^2G(\neg q)$

2. $L_2 = \{w : p \in w[i]\ \textit{for exactly three different}\ i \in \mathbb{N}\}$;
   (i) LTL - $\neg pU(p \wedge X(\neg pU(p \wedge X(\neg pU(p \wedge XG(\neg p))))))$
   (ii) PSL - another way to phrase $(\neg p)^* \cdot (p) \cdot (\neg p)^* \cdot (p) \cdot (\neg p)^* \cdot (p) \cdot G(\neg p)$

3. $L_3 = \{w : \textit{The cardinality of}\ \{i \in \mathbb{N} : p \in w[i]\}\ \textit{is finite and odd}\}$
   (i) LTL -

   Claim 0.5. *the language L_3 is not non-counting.*

   Proof. For every odd $n \in \mathbb{N}$. $p^n q^\omega \in L_3$ but, $p^{n+1}q^\omega \notin L_3$ □

   (ii) PSL - $(\neg((\neg p)^* \cdot p(\neg p)^* \cdot p \cdot (\neg p)^*)^*))G(\neg p)$

4. $L_4 = \{w : \textit{The cardinality of}\ \{i \in \mathbb{N} : p \in w[i]\}\ \textit{and}\ \{i \in \mathbb{N} : q \in w[i]\}\ \textit{are finite and equal}\}$;

   Claim 0.6. *L_4 is not is not definable by PSL formula, since $\mathbb{LTL} \subset \mathbb{PSL}$ therefore cannot defined by LTL.*

   Proof. we saw in class that $\mathbb{PSL} = \mathbb{NBW}$.
   Assume towards contradiction that we have an $\mathbb{NBW}$ B $s.t$ $[\![B]\!] = [\![L_4]\!]$.
   $\mathcal{B} = (\Sigma, Q, Q_0, \delta, F)$, since a NBW has a finite number of states,let $|Q| = n$. q, and p have to be finite but the word is infinite , thus there has to be another letter that repeats itself infinitely many times, let c be that letter. lets take a look at the word $w = p^{2n}q^{2n}c^\omega$. Easy to see that $w \in L_4$, we can notice that the prefix $w[..2n]$ which is $p^{2n} > n$ thus there has to be a state in B that repeat twice, let it be q_i. Let $\{q_{0_1}, .., q_{i_l}, ..., q_{i_k}, ..q_{j_{2n}}\}$ be the run for that prefix, now we can pump p^{2n} in w, to $w' = p^{2n+m(k-l)}q^{2n}c^\omega$. w' will repeat the loop from q_i to $q_j m$ times, continue with the same path of w and thus,eventually will accept too. It is clear that $w' \notin L_4$, in contradiction. Therefore L_4 is not $\mathbb{PSL}$ definable. □

Question 5

Let $\mathcal{C} = (\Sigma, Q, q_0, \delta, F)$ be a DCW such that $[\![\mathcal{C}]\!] = L \subseteq \Sigma^\omega$ for some alphabet $\Sigma = \{\sigma_1, \sigma_2, ..., \sigma_{|\Sigma|}\}$. We are to write an S1S formula $\psi_{\mathcal{C}}$ satisfying $[\![\psi_{\mathcal{C}}]\!] = L$. Let us denote the set of states of $\mathcal{C}$ as: $Q = \{q_1, q_2, ..., q_{|Q|}\}$. Since $\mathcal{C}$ is a DBW, it is deterministic and thus for every word $w \in \Sigma^\omega$, there is only one corresponding run which we will denote as ρ_w.
Let $w \in \Sigma^\omega$ be an input word to $\mathcal{C}$ and $\rho_w = q_{\rho_{w_0}} q_{\rho_{w_1}} q_{\rho_{w_2}} ...$ the only corresponding run of $\mathcal{C}$ on w. Let us define for w and for each state $q_i \in Q$ a corresponding bounded variable in a form of a set A_{w_i} that contains all the indices in which the run ρ_w passes at q_i, as in:

$$A_{w_i} = \{j \mid q_{\rho_{w_j}} = q_i\}$$

Since $q_{\rho_{w_0}} = q_0$, by definition we have: $\rho_{w_0} = 0 \in A_{w_0}$. Moreover, let us define for w and for each letter $\sigma_i \in \Sigma$ a corresponding set B_{w_i} that contains all the indices in which the word σ_i appears in w, as in:

$$B_{w_i} = \{j \mid w[j] = \sigma_i\}$$

Let us define a formula $\psi_{\mathcal{C}}$ as follows:

$$\begin{array}{lll}
 & \exists A_{w_1}, \exists A_{w_2} \cdots \exists A_{w_{|Q|}} & \\
\forall x & \left(\bigvee_{i=1}^{|Q|} x \in A_{w_i} \right) \wedge \bigwedge_{i=1}^{|Q|} \left(x \in A_{w_i} \rightarrow \bigwedge_{\substack{j=1 \\ j \neq i}}^{|Q|} x \notin A_{w_j} \right) & \wedge \\
\forall x & \left(\bigvee_{i=1}^{|\Sigma|} x \in B_{w_i} \right) \wedge \bigwedge_{i=1}^{|\Sigma|} \left(x \in B_{w_i} \rightarrow \bigwedge_{\substack{j=1 \\ j \neq i}}^{|\Sigma|} x \notin B_{w_j} \right) & \wedge \\
 & 0 \in S_0 & \wedge \\
\forall x & \bigvee_{(q_i, \sigma_j, q_k) \in \delta} x \in A_{w_i} \wedge x \in B_{w_j} \wedge S(x) \in A_{w_k} & \wedge \\
 & \bigwedge_{q_i \in F} \exists x \, \forall y \, (y \in A_{w_i} \rightarrow x > y) &
\end{array}$$

Let us provide an explanation for our construction:

1. The first line declares the existence of the bounded variables we defined earlier.

2. The second line corresponds to the fact that each natural number $x \in \mathbb{N}$ induces one position in the run; $x \in \rho_w$ and is associated with exactly one state $q \in Q$.

3. The third line corresponds to the fact that each natural number $x \in \mathbb{N}$ induces one position in the run; $x \in \rho_w$ and is associated with exactly one letter $\sigma \in \Sigma$.

4. The fourth line corresponds to fact we stated earlier that since $\mathcal{C}$ is deterministic, its initial state is singular - which we denoted as q_0.

5. The fifth line corresponds to fact that each natural number $x \in \mathbb{N}$ induces one element in the transition function δ.

6. The sixth line corresponds to the co-Büchi acceptance condition in that each state in the set F is reached a finite number of times.

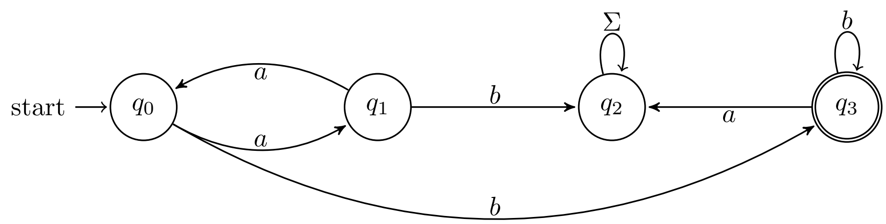
Question 6

(1)

In this section we are to compare the expressive power of LTL and DBW.

Claim 0.7. $\mathbb{DBW} \not\subseteq \mathbb{LTL}$

Proof. To prove so we'll present a language accepted by a DBW that cannot be accepted by an LTL formula. We saw in class that LTL cannot "count" - as in for every LTL-formula φ, the set $[\![\varphi]\!]$ is non-counting. Let $\Sigma = \{a, b\}$ and let $r = (aa)^* b^\omega$. We also saw in class that the language $[\![r]\!]$ is not non-counting. Therefore it cannot be accepted by an LTL formula. Let us construct a DBW $\mathcal{D}$ that accepts L:



One can see that $[\![\mathcal{D}]\!] = L$ and thus we constructed a DBW that accepts L. □

Claim 0.8. $\mathbb{LTL} \not\subseteq \mathbb{DBW}$

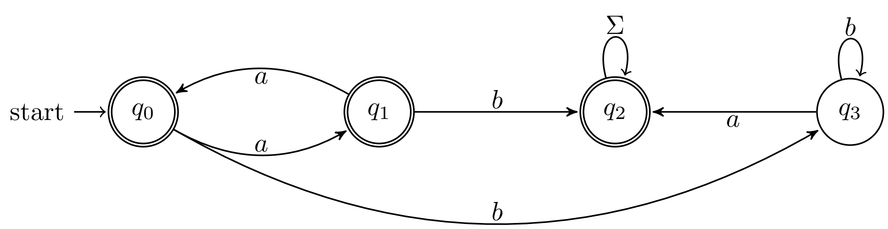
Proof. To prove so we'll present a language accepted by an LTL formula that cannot be accepted by an DBW. We saw in class that the language $L = \{w : |w|_a < \infty\}$ cannot be accepted by a DBW. Let $\varphi = (a \vee b)\ U\ G(b)$. One can see that $[\![\varphi]\!] = L$ and therefore we constructed an LTL formula that accepts L. □

(2)

In this section we are to compare the expressive power of LTL and DCW.

Claim 0.9. $\mathbb{DCW} \not\subseteq \mathbb{LTL}$

Proof. To prove so we'll present a language accepted by a DCW that cannot be accepted by an LTL formula. Let us define again $\Sigma = \{a, b\}$ and let $r = (aa)^* b^\omega$. Let us construct a DCW $\mathcal{C}$ that accepts L:



One can see that $[\![\mathcal{C}]\!] = L$ and thus we constructed a DBW that accepts L. □

Claim 0.10. $\mathbb{LTL} \not\subseteq \mathbb{DCW}$

Proof. To prove so we'll present a language accepted by an LTL formula that cannot be accepted by an DCW. By the DCW condition, we know that it cannot accept languages with words that have infinite conditions, so the language $L = \{w : |w|_a = \infty\}$ cannot be accepted by a DCW. Let $\varphi = G(a)$. One can see that $[\![\varphi]\!] = L$ and therefore we constructed an LTL formula that accepts L. □