Advanced Topics in Automata
Exercise 6

Submission: June 3, 2003

Exercise In this exercise we show that the *until* operator adds expressive power and that even with the until operator, LTL cannot count. In the following we make these intuitive sentences formal.

Very much like an automaton, an LTL formula defines a set of traces. Given an LTL formula φ over the set of propositions $Prop$ it defines a language over the alphabet $\Sigma = 2^{Prop}$. Formally, $models(\varphi) = \{\sigma \in \Sigma^\omega \mid \sigma, 0 \models \varphi\}$, namely, the set of traces over which φ is satisfied at time 0.

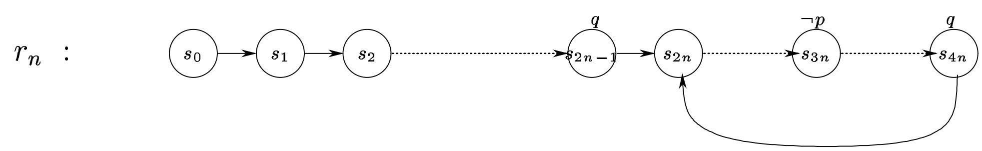

Figure 1: The structure r_n

1. We show that LTL without until (i.e. using only the temporal connectives $\Diamond$, $\Box$, $\bigcirc$) is not as strong as LTL (i.e. when adding $\mathcal{U}$) by giving two languages such that every LTL formula cannot separate the two languages. That is, given L_1 and L_2 for every LTL formula φ it cannot be the case that L_1 is contained in $models(\varphi)$ and L_2 does not intersect $models(\varphi)$ (and vice versa). It must be the case that either φ holds for some word in L_1 and for some word in L_2 or does not hold for both.

   Consider the system r_n in Figure 1. We assume that all states not labeled by q are labeled by $\neg q$, and all the states not labeled by $\neg p$ are labeled by p. Let v_n denote the trace from state s_{2n} (i.e. $[(p, \neg q)^n$ $(\neg p, \neg q)$ $(p, \neg q)^{n-2}$ $(p, q)]^\omega$), and let u_n denote the trace from state s_0 (i.e. $(p, \neg q)^{2n-1}(p, q)u_n$).

Clearly for u_n and v_n there exists a simple LTL formula that distinguishes the two (at time 0). This formula should only require that p holds after n time units (i.e. $\varphi = \bigcirc^n p$). Let $L_1 = \bigcup_{i=1}^{\infty} u_i$, and $L_2 = \bigcup_{i=1}^{\infty} v_i$.

Show by induction on the structure of the formulas that for every formula φ there exists a large enough i such that $u_i \models \varphi$ iff $v_i \models \varphi$. It is easy to see that for every n we have $v_n \models p\mathcal{U}q$ and $u_n \not\models p\mathcal{U}q$.

The intuition is that with $\Box$ and $\Diamond$ the two different languages are synchronized and we loose the ability to distinguish between them. The only way to distinguish is by using $\bigcirc$. However, $\bigcirc$ helps only in the prefix and once i is large enough the formula fails to distinguish. Hence, your proof should take into account the number of $\bigcirc$ operators in φ.

r_n :

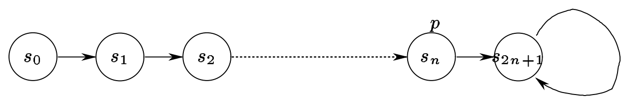

Figure 2: The structure r_n

1. We show that LTL cannot count. In a similar way to the above, we give two languages such that every LTL formula cannot separate the two. Consider the system r_n in Figure 2. We assume that all states not labeled by p are labeled by $\neg p$. We abuse notation and denote by r_i also the trace of r_i. Namely, $r_i = (\neg p)^i p (\neg p)^\omega$.

   Let $L_e = \bigcup_{i=0}^{\infty} r_{2i}$ and $L_o = \bigcup_{i=1}^{\infty} r_{2i+1}$. Show by induction on the structure of the formulas that for every formula φ there exists a large enough i such that $r_i \models \varphi$ iff $r_{i+1} \models \varphi$. It is quite simple to construct an NBW that accepts L_e (similarly for L_o).

   Again, the number of $\bigcirc$ operators in φ can help distinguish between prefixes. Once i is large enough, the number of $\bigcirc$ operators is not usefull.