SS 2019 07.06.2019

Exercises to the lecture
Algorithmic Automata Theory
Sheet 6

Dr. Prakash Saivasan Delivery until 18.06.2019 at 15:00

Exercise 6.1 (Presburger formulas & Parikh images)

1. Present a Presburger formula φ such that every bound variable occurs in *precisely* one atomic expression and such that

$$Sol(\varphi) = \left\{ \begin{pmatrix} 2n+1 \\ n+3 \end{pmatrix} \middle| \; n \in \mathbb{N} \right\} \cup \left\{ \begin{pmatrix} 3n+1 \\ 2n+2 \end{pmatrix} \middle| n \in \mathbb{N} \right\} \; .$$

2. The *Parikh image* $\Psi : \Sigma^* \to \mathbb{N}^\Sigma$ mapps each word w to the vector $\Psi(w)$, where the components store the number of occurrences of each letter in w. For a language $\mathcal{L} \subseteq \Sigma^*$, let $\Psi(\mathcal{L}) = \{\Psi(w) \mid w \in \mathcal{L}\}$. For example for $\Sigma = \{a, b, c\}$:

$$\Psi(ababcb) = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} \text{ and } \Psi((aa)^*(bbb)^*) = \left\{ \begin{pmatrix} 2n \\ 3m \\ 0 \end{pmatrix} \middle| n, m \in \mathbb{N} \right\} \; .$$

Give an NFA A so that $\Psi(\mathcal{L}A) = Sol(\varphi)$ for the Presburger formula φ from a).

Exercise 6.2 ("Presburger $\Rightarrow$ NFA"for atomic formulas with equality)
One can modify the algorithm for $\vec{a}\,\vec{x} \leq b$ to produce an NFA for $\vec{a}\,\vec{x} = b$ by making the state $0 \in \mathbb{Z}$ the only accepting state and by changing the transition relation so that a transition

$$q \xrightarrow{\vec{\beta}} \frac{1}{2}(q - \vec{a}\,\vec{\beta})$$

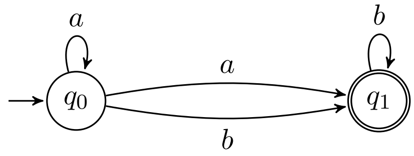
is only added if $q - \vec{a}\,\vec{\beta}$ is even.

1. Use the modified algorithm to construct a finite automaton for $x - 2y = 1$.

2. Verify your result in 1 by checking that

$$\mathcal{L}A_{x-2y=1} = \mathcal{L}A_{x-2y\leq 1} \cap \mathcal{L}A_{-x+2y\leq -1} \; .$$

Exercise 6.3 (NBA Complementation)
Consider the NBA A over $\Sigma = \{a, b\}$ below:



Use Büchi's complementation method discussed in class to compute $L(A)$ and $\overline{L(A)}$.

Exercise 6.4 (Muller Automata)
A Nondeterministic Muller Automaton (NMA) is a tuple $A = (Q, \Sigma, \delta, q_0, F)$. The first four components are as in Büchi automata. $F = \{Q_F^1, \ldots, Q_F^n\} \subseteq \mathcal{P}(Q)$ is a set of sets of states instead of a single set of states. The idea is to accept a run if the set of states that occur infinitely often matches one of the Q_F^i exactly. Formally, a run r of A is accepting if $\mathit{Inf}(r) \in F$ where $\mathit{Inf}(r)$ is the set of states that are visited infinitely often in r. As for Büchi automata, we call A a Deterministic Muller Automaton (DMA) if for each $q \in Q$ and $a \in \Sigma$ there is exactly one state $q' \in Q$ such that $(q, a, q') \in \delta$.

1. Given an NBA A, show that there is an NMA A_{NMA} such that $L(A_{NMA}) = L(A)$.
2. Show that DMA are strictly more expressive than DBA.
3. Given a DMA A, show that there is an NBA A_{NBA} such that $L(A_{NBA}) = L(A)$.
4. Prove that DMA are closed under complement, i.e. for every DMA A there exists a DMA $\bar{A}$ with $L(\bar{A}) = \overline{L(A)}$.

Delivery until 18.06.2019 at 15:00 into the box next to 343 or in the class