Problems for Automata, Logic and Games

November 26, 2021

1 Finite words and semigroups

1. **Problem.** Consider models with no structure except equality. Show that over such models, first order logic has the same expressive power as monadic second order logic.

2. **Problem.** Consider models with no structure except equality. Show that if a first order formula of quantifier rank k is true in some model, then it is true in a model with $O(k)$ elements. Show also that this is not true for MSO formulas.

3. **Problem.** Consider two ways of representing a finite word as a model: we either have the order relation $x < y$ (which is our default choice for the lecture), or we have the successor relation $x = y + 1$. Show that for both ways, MSO gives the same expressive power.

4. **Problem.** Consider finite words, represented using order $x < y$. Show that every MSO formula is equivalent to a formula of the form $\exists X \varphi(X)$ where $\varphi(X)$ is a first-order formula. In other words, a single existential quantifier, on the exterior of the formula, is enough. Hint: convert the formula into an automaton, and then express "there is an accepting run", by describing only the states in some positions, say every k-th position, where k depends only on the automaton.

5. **Problem.** Show a sequence of languages such that: (a) the n-th language is recognised by a deterministic automaton with $\mathcal{O}(n)$ states; and (b) the reverse of the n-th language is recognised by a deterministic automaton with $\mathcal{O}(n)$ states; and (c) the smallest monoid recognising the language n-th language has size exponential in n.

6. **Problem.** Prove that for every regular language there is an optimal monoid homomorphism: i.e. a homomorphism $h : \Sigma^* \to M$ which recognises the language, and such that if $g : \Sigma^* \to N$ is another monoid homomorphism that recognises the language, then h stores less information than g in the sense that

$$g(w) = g(v) \qquad \Rightarrow \qquad h(w) = h(v)$$

holds for every words $w, v \in \Sigma^*$.

7. **Problem.** Prove that an example of the optimal homomorphism from the previous exercise is the function that maps a word w to the state transformation that it induces in the minimal deterministic automaton.

8. **Problem.** Recall the two representations from Exercise 3 of Lecture 1. Show that they are not equivalent for first-order logic, because there is a property of finite words that can be described in first-order logic with $x \leq y$, but cannot be described in first order logic with $x = y + 1$.

9. **Problem.** Show that (non-monadic) second order logic is not compositional, i.e. the rank k theory of two words w and v does not uniquely determine the rank k theory of their concatenation wv.

2 First-order logic and friends

1. **Problem.** Write a star-free expression that defines the language $(ab)^*$.

2. **Problem.** Consider the word languages that can be defined in first-order logic without the order predicate $x < y$, but only using the label predicates $a(x)$ for $a \in \Sigma$ (and equality). Show that a language belongs to this class if and only if it is recognised by a finite monoid that is both aperodic and commutative (commutative means that $a \cdot b = b \cdot a$).

3. **Problem.** Consider the variant of LTL where Until is removed, but we add F (finally). For example, the following formula defines the language of words that have length exactly 5:

$$\underbrace{(\mathsf{FFFF}\mathrm{true})}_{\text{length at least 5}} \qquad \wedge \qquad \underbrace{\neg(\mathsf{FFFFF}\mathrm{true})}_{\text{length at most 6}}.$$

   Show that this variant cannot express "the second position has label a".

4. **Problem.** Consider a two-way extension of LTL, which adds a backward operator called "since". The formula β Since α holds in a position x if there exists some position $y < x$ where α holds, and such that β holds in every position z that satisfies $y < z < x$. Show that for every formula in the two-way extension, the formula is equivalent over finite words to a Boolean combination of one-way formulas (a formula is one-way if it avoids Until, or it avoids Since). This result is known as Gabbay's separation theorem (it also works beyond finite words).

3 Automata for ω-words

1. **Problem.** Show that languages recognized by nondeterministic Büchi automata are closed under intersection. Give a direct construction, which does not pass through equivalence with other models.

2. **Problem.** Show that if a nondeterministic Büchi automaton is nonempty (accepts at least one ω-word), then it accepts some that is *ultimately periodic*, i.e. it has the form uv^ω for some finite words u and v.

3. **Problem.** Consider safety automata for ω-words, i.e. automata where every run is accepting (as in slide 5 of Lecture 4). Show that the language "a infinitely often" is not recognized by a finite Boolean combination of safety automata. Usually this is left withoug saying, but in the context of the next two problems, we underline that the automata have finite state spaces. Hint: recall that safety automata recognize closed languages, and therefore if a safety automaton reject an ω-word, then there is a finite prefix which gurantees rejection.

4. **Problem.** Show that every language of ω-words is recognized by some nondeterministic safety automaton with a possibly infinite state space.

5. **Problem.** Show that the language "infinitely often a" is recognized by a nondeterministic safety automaton with a countably infinite state space.

6. **Problem.** A co-Büchi automaton is one where the acceptance condition is given by a distinguished subset of transitions, and a run is accepting if it uses transitions from the subset finitely often. Show that a nondeterministic co-Büchi automata cannot recognize the language "infinitely often a". (Hint. Consider cycles that read b letters and avoid distinguished transitions.)

4 Determinism and alternation

1. **Problem.** Consider alternating automata on finite words. Show that even if the input alphabet has one letter only, then the conversion of an alternating automaton into a nondeterministic one is not polynomial (i.e. there is no polynomial p such that every alternating automaton on a one letter alphabet with n states can be converted into an equivalent nondeterministic one with at most $p(n)$ states).

2. **Problem.** Consider parity games with only one priority 1, and ranks as defined on slide 22 of Leture 6. Show that for every ordinal number α there is a game where the initial position has this rank.

3. **Problem.** Consider parity games with only one priority 1, and ranks as in the previous problem. Show that if the game is possibly infinite, but finitely branching (i.e. every position has finitely many outgoing edges), then for every position the rank is either undefined, or a natural number.

4. **Problem.** Consider an alternating parity automaton on ω-words with possibly more priorities than 1. For an input ω-word, acceptance is defined in terms of a parity game. For this game, we can consider ranks,

these will now concern the maximal priority. Show an example of an alternating automaton and an input word, such in the initial position of the corresponding parity game, the rank is ω.

5 Trees

1. **Problem.** Show that the emptiness problem for nondeterministic tree automata on finite trees is complete for polynomial time, with respect to LogSpace reductions.

2. **Problem.** Show that the universality problem for tree automata for nondeterministic tree automata on finite trees complete for ExpTime. **Hint.** For the lower bound, reduce from the following problem, which is known to be complete for ExpTime: given an alternating Turing machine and a number n in unary, decide if the machine can accept the empty word using memory at most n. In an alternating Turing machine, the states have owners from $\{\forall, \exists\}$, and the player who owns the state chooses the next transition. To accept a word, player $\exists$ must have a winning strategy that ensures that an accepting state is reached in finite time.

3. **Problem.** Consider MSO on finite trees. Show that MSO is equivalent to the fragment where set quantification is restricted to subsets of the leaves. **Hint.** Every set of nodes in a binary tree can be represented using two subsets of leaves.

4. **Problem.** Give a nondeterministic parity automaton on infinite trees over alphabet $\{a, b\}$, which recognizes the language "for every node, there is a descendant with label a".

6 Fixpoints and Kripke structures

1. **Problem.** Show that for every formula $\varphi(X, Y)$ of the modal μ-calculus with two free variables, the following formulas are equivalent:

$$\mu X \mu Y \varphi(X, Y) \qquad \mu Y \mu X \varphi(X, Y).$$

2. **Problem.** Prove that the MSO theory is decidable for the infinitely branching tree $\mathbb{N}^*$ equipped with the ancestor relation only.

3. **Problem.** Consider an alternating automaton on Kripke structures that has n states. Show that if the automaton accepts some structure, then it accepts one where every vertex has at most n outgoing edges.

4. **Problem.** Show that the following problem is decidable: given an alternating automaton on Kripke structures, decide if it accepts some Kripke structure. **Hint.** Use tree unravelling.

5. **Problem.** Show that if an alternating automaton on Kripke structures accepts some structure, then it accepts a finite one. **Hint.** Regular trees are finite up to bisimilarity.