Assignment 2: Turing Machines

John Pennell
ID: 834035412
The University of Auckland

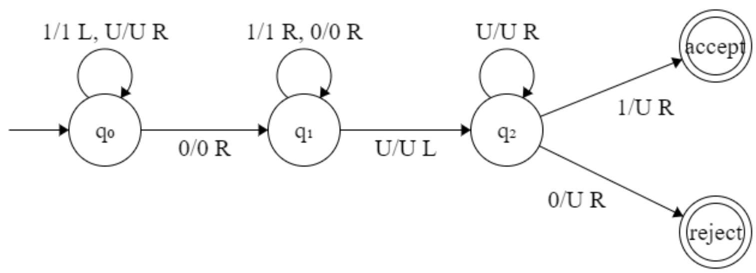
May 19, 2017

Question One

a) Let the definition be stated: A Turing machine M with input alphabet S is a decider if, for all strings w over S, M either accepts or rejects w.

In order to prove that M is not a decider, we must show that there is at least one string that M can neither accept or reject. Consider the string input 1, we see that the machine cannot halt due to a loop in the start state q_0, which has the transition $\delta(q_0, 1) = (q_0, 1, L)$, continuously staying in the same position as the head cannot read any further to the left of the input, thus never halting at either q_{accept} or q_{reject}. A configuration of this transition shows this as $q_0 1 \vdash_M q_0 1 \vdash_M$ ..., which inevitably runs forever. This as a result, contradicts the stated definition at the start of this proof, thus M is not a decider.

b) Let $L = \{0u1 \mid u \in \Sigma^*\}$, where $\Sigma = \{0, 1\}$ and $\Gamma = \{0, 1, \sqcup\}$.



Consider the Turing Machine depicted above ($U = \sqcup$):

Proving $L \subseteq L(M)$:
To show that $L \subseteq L(M)$, let $w = w_1, w_2...w_n$ and suppose that $w \in L$. Then w should start with a 0 and end with a 1. This is proven with the Turing machine M displayed above, staying in q_0 until the first zero is seen in the given input tape (proven in question 1a). The machine also never halts if the input string is the empty string ϵ, as it will stay in q_0 due to the transition $\delta(q_0, \sqcup) = (\sqcup, R)$, considering that it is continuously $\sqcup$ to the right side of the input tape. In state q_1 we see that it takes in input Σ^* until it hits the end of the string which is $\sqcup$. From there the head moves back one and into state q_2. In state q_2 we can either enter q_{accept} if the last symbol is a 1, or q_{reject} if it is a 0.

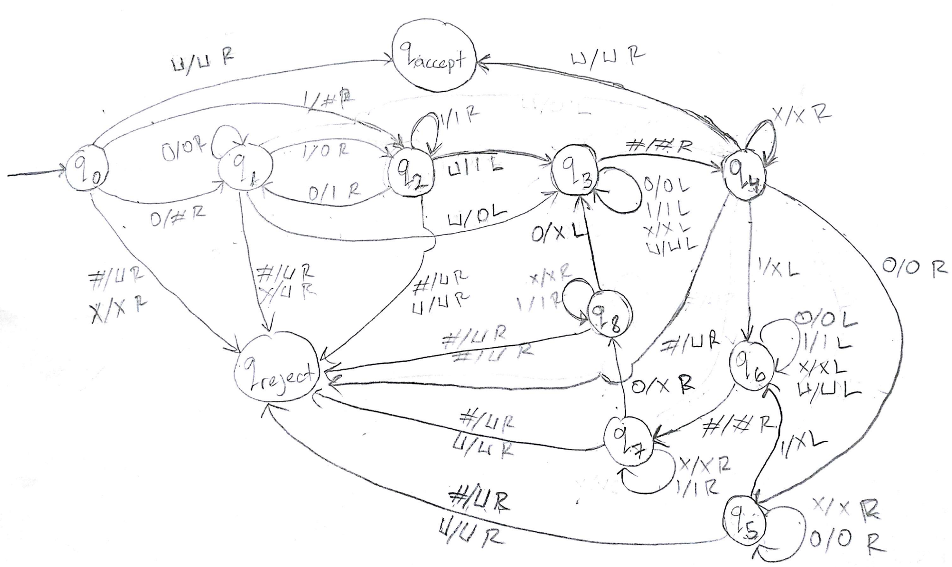
Proving $L(M) \subseteq L$:
Now to show that $L(M) \subseteq L$, consider $w \in L(M)$ taken from the first part of this proof. There is a sequence of configurations (cf for short) $C_1, C_2, ..., C_{k+1}$, such that C_1 is the start cf of of M on w, each $C_i \vdash_M C_{i+1} (for \leq i \leq k \leq n)$, and C_{k+1} as the accepting cf.

Suppose that $C_{k+1} = uq_{accept}v$, such that $u \in \Sigma^*$ and ends with a 1, and $v \in \epsilon$. The transition that yields $C_k + 1$ from the preceding cf C_k is $\delta(q_2, 1) = (q_{accept}, \sqcup, R)$ as that is the only place q_{accept} occurs in δ. It follows that the last symbol of u is a 1, if the transition for q_1 is $\delta(q_1, 1) = (q_1, 1, R)$ such that $u = z1$ for some string $z \in 0, 1*$ and that $C_k = zq_1 1v$. Ergo $w \in L$.

c) The language L is Decidable iff there is a Turing Machine M which will accept strings in the language and reject strings not in the language. It is obvious that L is Turing decidable since there are strings that the machine can and cannot accept from the language L, i.e. consider an input tape w = 01, it is obvious that the Turing machine above will accept the string. Also with an input w = 00, the machine simply rejects the string. We also see that state q_1 transitions to itself, since $u \in \Sigma^*$, where the machine must halt at the followed states. Thus L is decidable.

Question Two

a) **(i)** Consider $L = \{w \in \{0,1\}^* \mid the\ number\ of\ 0s\ in\ w\ is\ exactly\ twice\ the\ number\ of\ 1s\}$. Then, let $M = \{Q, \Sigma, \delta, \Gamma, q_s, q_{accept}, q_{reject}\}$, such that $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup, X, \#\}$ and $q_s = q_0$.



An implementation-level description of M on input w:

M = on input w:
1. Scan the tape input w until a 1 is found, mark it out with the symbol 'X' and move the head of the tape back to the left-hand end, and then go to stage 2. If there is no 1s and 0s left, go to stage 4, otherwise go to stage 5.

2. Scan the tape until a 0 is found, cross it out and go to stage 3. Otherwise, if there is no 0s left, go to stage 5.

3. Continue scanning w until another 0 is found, cross it out and move the head back to the left-hand end of the tape, and go to stage 1. If there are no 0s left, go to stage 5.

4. Accept.

5. Reject.

(ii) Here are the following sequence trace via the configuration:

Empty string: $q_0 \sqcup \vdash_M \sqcup q_{accept}$

01: $q_0 01 \vdash_M \#q_1 1 \vdash_M \#0q_2 \vdash_M \#q_3 01 \vdash_M q_3\#01 \vdash_M \#q_4 01 \vdash_M \#0q_5 1 \vdash_M \#q_6 0x \vdash_M q_6\#0x \vdash_M \#q_7 0x \vdash_M \#xq_8 x \vdash_M \#xxq_8 \vdash_M \#xxq_{reject}$

010010: $q_0 010010 \vdash_M \#01001q_3 \vdash_M {}_3\#010010 \vdash_M \#q_4 010010 \vdash_M \#0q_5 10010 \vdash_M q_6\#0X0010 \vdash_M \#q_7 0X0010 \vdash_M \#XXq_8 0010 \vdash_M q_3\#XXX010 \vdash_M \#XXXq_4 010 \vdash_M \#XXX0q_5 10 \vdash_M \#XXXq_6 0X0 \vdash_M \#q_7 XXX0X0 \vdash_M \#XXXXq_7 0x0 \vdash_M \#XXXXXXq_8 0 \vdash_M q_3\#XXXXXXXX \vdash_M \#XXXXXXXXq_{accept}$

b) Suppose B is Turing-decidable, then let M decide B. We can show M_C, since it's simply the compliment of M, where M_C decides the complement B_C of B. To get M_C, we form a Run M on input w. We swap the final states, such that if M accepts, reject and if M rejects, accept. It is obvious that M_C is a decider because M is, and M_C accepts w iff M rejects w iff w is in B_C. Ergo B is Turing decidable, since we see that M_C decides compliment of B.

Question Three

To prove that L is Turing recognizable, we consider that there is a Turing machine M, such that M recognizes L.

Since A is a DFA and $L(A) \cap \overline{L(B)} = \oslash$, we convert NFA B to its equivalent DFA B'. So, $L(A) \cap \overline{L(B)}$ is still $\oslash$. We then run M on input w, where

$M =< A, B', w >$. If both A and B' halts on input w, we can say that the language L is recognizable, since it follows that any Turing-decidable language is also Turing- recognizable.

Since the set of recognizable languages is closed under complement, it is obvious that the compliment of the languages L is also Turing recognizable.

Question Four

Since $w \in A$ and $w \in C$, then $w \in (A \cup C)$, also since $w \notin B$ and $w \notin D$, then $w \notin (B \cup D)$.

Let $X = A \cup C$ and $Y = B \cup D$, such that $w \in X$ and $w \notin Y$.

Suppose X and Y are Turing decidable, then there exists machines M_1 and M_2, such that M decides the set of the languages X and Y on input w. If machine M_1 accepts w and machine M_2 rejects w, then it is clear that M accepts w. Otherwise, M rejects w. Since the Turing machines are decidable, we know that on any given input the machine M must halt at either reject or accept states. Thus, the set of Turing-decidable languages is closed under Θ.

We see that the sets of Θ is Turing decidable, however is not Turing recognizable, we know this to be the halting problem, where the language is unrecognizable. In order to prove this we must construct a Turing machine M_H, such that a machine M halts on input $< M, w >$. Suppose we have constructed this machine, such that $w \in \Theta$. We will then end up with a machine that will halt on its own input, which is indeed decidable, however by construction it does not accept. This is a contradiction, where it follows that the machine is unrecognizable.

Thus the set of Turing-decidable languages are closed under Θ, but the sets of Turing-recognizable languages is not closed under Θ.