Infinite Automata 2025/26

Lecture Notes 1

Henry Sinclair-Banks

Recall from the Exercise Sheet 1.

Theorem 1.1. Reachability in pushdown automata is decidable in polynomial time.

Definition 1.2. A *one-counter machine (1-CM)* consists of a finite set of control states Q and a set of transition $T \subseteq Q \times \{-1, 0, +1, =0?\} \times Q$; denoted (Q, T). The counter must remain nonnegative at all times, so a configuration of a 1-CM comprises of a control state $q \in Q$ and a counter value $x \in \mathbb{N}$; denoted (q, x).

Defintion 1.2. Reachability in 1-CMs (problem).

Input. A 1-CM M, an initial configuration $(s, 0)$, and a target configuration $(t, 0)$.

Question. Does there exist a run from $(s, 0)$ to $(t, 0)$ in M?

We will also use the notation $(p, 0) \xrightarrow{*}_M (q, 0)$ to denote the existence of a run from $(p, 0)$ to $(q, 0)$ in M.

Theorem 1.3. Reachability in 1-CMs is decidable in polynomial time.

Proof sketch. Construct a pushdown automata P that simulates a given 1-CM M. The height of the stack (minus one) will equate to the counter value. Let $\Gamma = \{\$, a\}$ be the stack alphabet. At the bottom of the stack, we will place one '\$'. The number of 'a's on top of the '\$' will correspond to the counter value of the 1-CM.

- If $(p, 0, q)$ is a transition in M, there will be a transition (p, ε, q) in P.
- If $(p, +1, q)$ is a transition in M, there will be a transition $(p, \text{push}(a), q)$ in P.
- If $(p, -1, q)$ is a transition in M, there will be a transition $(p, \text{pop}(a), q)$ in P.
- If $(p, =0?, q)$ is a transition in M, there will be a pair of transitions $(p, \text{pop}(\$), p')$ and $(p', \text{push}(\$), q)$ in P.

It follows that $(s, 0) \xrightarrow{*}_M (t, 0)$ if and only if $(s, \$) \xrightarrow{*}_P (t, \$)$. Hence we can use Theorem 1.1 to decide reachability in 1-CMs in polynomial time. □

Definition 1.4. A *logarithmic space* Turing machine M is a Turing machine with the following properties. There are two tapes: one tape is a read-only input tape and the other is a read-write work tape. There there exists a constant $c \in \mathbb{N}$ such that, on input $x \in \{0, 1\}^*$, M halts (and accepts or rejects) whilst the work tape head never exceeds $c\lceil \log(n) \rceil$. In other words, the size of the work tape is bounded by $\mathcal{O}(\log(n))$.

Definition 1.5. NL (complexity class). A problem X belong to NL if there exist a *non-deterministic logarithmic space* Turing machine M such that, on input $x \in \{0, 1\}^*$, M halts and accepts if $x \in X$, otherwise M halts and rejects if $x \notin X$.

Fact 1.6. $\mathsf{NL} \subseteq \mathsf{P}$.

Proof sketch. Given a non-deterministic logarithmic space Turing machine M and an input string $x \in \{0,1\}^*$, construct a directed graph $G = (V, E)$ where the vertices $V = \{\text{all configurations of } M\}$ and E is defined as follows. Suppose there are two configuration c_1 and c_2 such that there is a single transition a in M such that $c_1 \xrightarrow{a} c_2$, then $(c_1, c_2) \in E$. In other words, the edges of G correspond to what M can do using just one transition.

This construction can be complete in polynomial time because the number of possible configurations of M is bounded above by the product of the following values.

- $|x|$ for the head position over the input tape.
- $c\lceil \log |x| \rceil$ for the head position over the work tape.
- $2^{c\lceil \log |x| \rceil}$ for the contents of the work tape (assuming that the alphabet of the work tape has cardinality 2).
- $|Q|$ for the current control state.

Further, we add one additional final node to the graph f. We also add some final edges to the graph. Let c be an arbitrary configuration of M at the "halt and accept" state, then we will add the edge $(c, f) \in E$.

Suppose that i is the initial configuration of the M, it follows that M halts on and accepts input x if and only if $i \xrightarrow{*}_G f$. We can therefore decide whether M halts on and accepts input x by constructing G and deciding $i \xrightarrow{*}_G f$. This last step can trivially be completed in polynomial time using BFS or DFS. □

Definition 1.7. Directed graph reachability (problem).

Input. A directed graph G, an initial node s, and a target node t.

Question. $s \xrightarrow{*}_G t$?

Theorem 1.8. Directed graph reachability is NL-complete.

Proof sketch. First, NL-hardness follows from the arguments presented in the proof sketch of Fact 1.6.

Second, we will argue directed graph reachability is in NL. Consider the following non-deterministic algorithm for directed graph reachability. Consider an arbitrary directed graph $G = (V, E)$, an initial node s, and a target node t. Let $n = |V|$.

1. Let $v \leftarrow s$. *Set the current node to the starting node.*
2. Let $\ell \leftarrow 1$. *Set the current path length to one.*
3. While $\ell \leq n$:
   (a) If $v = t$, halt and accept.
   (b) Among the neighbours of v, non-deterministically select a new current node $v \leftarrow v'$.
   (c) $\ell \leftarrow \ell + 1$.
4. Halt and reject. *If t could not be reached in n steps, then t cannot be reached from s.*

It remains to argue that this non-deterministic algorithm runs in logarithmic space. There are only two variables to maintain: v and ℓ. First, the value of ℓ is between 1 and $|V|$, so ℓ can be stored in $\lceil \log(n) \rceil$ space using binary encoding. Similarly, we can number the vertices $1, 2, \ldots, n$ and v can store the number of a given vertex and so v can also be stored in $\lceil \log(n) \rceil$ space using binary encoding. □

Theorem 1.9. Reachability in 1-CMs is in NL.

Lemma 1.10. Let M be a 1-CM and let $(p, 0)$, $(q, 0)$ be two configurations. Let n be the number of states in M. There exist a polynomial f such that if $(p, 0) \xrightarrow{*}_M (q, 0)$, then there exist a run from $(p, 0)$ to $(q, 0)$ such that all configurations in the run have counter values at most $f(n)$.

Proof. Let $(p, 0) \xrightarrow{\pi} (q, 0)$ be the run (in M) which, among all other runs, has the least greatest counter value. Let (r, x) be the configuration in $(p, 0) \xrightarrow{\pi} (q, 0)$ with the greatest counter value. For the sake of contradiction, suppose that $x > 2n^2 + 2n$.

For convenience, suppose π_1 and π_2 are the prefix and suffix of π such that $(p, 0) \xrightarrow{\pi_1} (r, x) \xrightarrow{\pi_2} (q, 0)$. We will now examine $(p, 0) \xrightarrow{\pi_1} (r, x)$ in detail; symmetric arguments can be applied to $(r, x) \xrightarrow{\pi_2} (q, 0)$. Let $q_i(i)$ be the *last* configuration in $(p, 0) \xrightarrow{\pi_1} (r, x)$ with the counter value i. We shall call these configurations *marked configurations.*

Consider the n^2+n marked configurations $q_{n^2+n+1}(n^2+n+1), q_{n^2+n+2}(n^2+n+2), \ldots, q_{2n^2+2n}(2n^2+2n)$. We will group these marked configurations in n blocks, each consisting of $n+1$ marked configurations.

- Block 1: $q_{n^2+n+1}(n^2 + n + 1), q_{n^2+n+2}(n^2 + n + 2), \ldots, q_{n^2+2n+1}(n^2 + 2n + 1)$.
- Block 2: $q_{n^2+2n+2}(n^2 + 2n + 2), q_{n^2+2n+3}(n^2 + 2n + 3), \ldots, q_{n^2+3n+2}(n^2 + 3n + 2)$.
- $\cdots$
- Block n: $q_{2n^2+n}(2n^2 + n), q_{2n^2+n+1}(2n^2 + n + 1), \ldots, q_{2n^2+2n}(2n^2 + 2n)$.

Now, using pigeonhole principle, observe that a cycle can be found in every block. Since there are $n+1$ marked configurations in a given block, there must be two marked configurations $q_i(i)$ and $q_j(j)$ with the same state $q_i = q_j$. Let $q = q_i = q_j$. Accordingly, the run from $q(i)$ to $q(j)$ is a cycle that adds $j - i$ to the counter. Importantly, observe that $1 \leq j - i \leq n$.

End of lecture, to be continued.