1 Infinite Computations

Theorem 1. *NBA-recognizable languages are closed under complement.*

Proof. Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ be an NBA. We define the transition profile of a word $w \in \Sigma^*$ as a finite directed graph $\mathrm{t}(w) = (Q, E, E_F)$ with $E = \{(p,q) \in Q \times Q \mid \mathcal{A} : p \xrightarrow{w} q\}$ and $E_F = \{(p,q) \in Q \times Q \mid \mathcal{A} : p \xrightarrow[F]{w} q\}$. Then we define $\mathrm{TP} = \{\mathrm{t}(w) \mid w \in \Sigma^*\}$ and the transition profile automaton $\mathrm{TPA}_F = (\mathrm{TP}, \Sigma, \mathrm{t}(\varepsilon), \delta_{\mathrm{TPA}}, F)$ with $\delta_{\mathrm{TPA}}(\mathrm{t}(u), a) = \mathrm{t}(ua)$.

For $t \in \mathrm{TP}$, let $U_t = \{u \in \Sigma^+ \mid \mathrm{t}(u) = t\}$. These sets are regular, as they are accepted by the NFA $\mathrm{TPA}_{\{t\}}$.

Let $\bar{t} = t_0 t_1 \cdots \in \mathrm{TP}^\omega$ be an infinite sequence of transition profiles. We call $\bar{t}$ accepting if there are $q_1 \cdots \in Q^\omega$ such that every t_i has an edge (q_i, q_{i+1}) and infinitely many of these edges are labeled by F. Let $\mathrm{NTP} = \{(t_0, t_1) \in \mathrm{TP} \times \mathrm{TP} \mid t_0 t_1^\omega \text{ is non-accepting}\}$.

Claim : $L(\mathcal{A})^\complement = \bigcup_{(t_0,t_1) \in \mathrm{NTP}} U_{t_0} U_{t_1}^\omega$. Then one can construct an NBA for $L(\mathcal{A})^\complement$.

For every $\alpha \in \Sigma^\omega$, let $\alpha = u_0 u_1 \cdots \in (\Sigma^*)^\omega$ be a factorization of α into finite words and let $\bar{t} = \mathrm{t}(u_0)\mathrm{t}(u_1)\ldots$. Then $\bar{t}$ is accepting iff $\alpha \in L(\mathcal{A})$.

In particular, this is true for $\mathrm{t}(u_1) = \mathrm{t}(u_2) = \ldots$, so it remains to be shown that every word α has such a factorization.

Ramsey's Theorem: Let C be a finite set, X with $|X| = \aleph_0$, $E = \{(x,y) \in X \times X \mid x \neq y\}$, and $f : E \to C$. Then there is an infinite $Y \subseteq X$ such that $|f(E \cap (Y \times Y))| = 1$.

For a $\alpha \in \Sigma^\omega$, let $X = \mathbb{N}$ and $C = \mathrm{TP}$. For $i < j$ we set $f(j,i) = f(i,j) = \mathrm{t}(\alpha[i,j])$. Using Ramsey's theorem, there is an infinite $Y \subseteq \mathbb{N}$ such that $f(E \cap (Y \times Y)) = \{t\}$. Let $Y = \{i_j \mid j > 0\}$ such that $i_j < i_{j+1}$ and $i_0 := 0$. We define $u_j = \alpha[i_j, i_{j+1}]$. Then $\alpha = u_0 u_1 u_2 \ldots$ and $\mathrm{t}(u_1) = \mathrm{t}(u_2) = \ldots$. □

Theorem 2. *For every $\varphi \in LTL$, one can construct an equivalent GBA with $\mathcal{O}(2^{|\varphi|})$ states.*

Proof. Let $\mathrm{cl}(\varphi) \subseteq \mathrm{LTL}$ be the set of all sub-formulas in φ (including φ itself). We define a φ-expansion of a word $\alpha \in (\mathbb{B}^n)^\omega$ as a function $\beta : \mathbb{N} \times \mathrm{cl}(\varphi) \to \mathbb{B}$ as below. The GBA then is $\mathcal{A} = (Q, \mathbb{B}^n, q_0, \Delta, F)$.

- $Q = \{q_0\} \cup 2^{\mathrm{cl}}(\varphi)$
- $F = (F_\psi)_{\psi \in \mathrm{Ucl}}$ where $\mathrm{Ucl} = \{\psi \in \mathrm{cl}(\varphi) \mid \psi = \psi_1 U \psi_2\}$
- $\Delta : q_0 \xrightarrow{a} \Phi$ with $\varphi \in \Phi$ according to φ-expansion rules
- $\Delta : \Phi \xrightarrow{a} \Psi$ according to φ-expansion rules

β is a φ-extension of α if it satisfies the following local conditions

- $\beta(i, p_j) = a_j$ where $\alpha(i) = (a_1, \ldots, a_n)$.
- $\beta(i, \neg\psi) = 1$ iff $\beta(i, \psi) = 0$.

- $\beta(i, \psi \wedge \vartheta) = 1$ iff $\beta(i, \psi) = \beta(i, \vartheta) = 1$.
- $\beta(i, \psi \vee \vartheta) = 1$ iff $\beta(i, \psi) = 1$ or $\beta(i, \vartheta) = 1$.
- $\beta(i, X\psi) = 1$ iff $\beta(i+1, \psi) = 1$.
- $\beta(i, \psi U \vartheta) = 1$ iff $\beta(i, \vartheta) = 1$ or $[\beta(i, \psi_1) = 1$ and $\beta(i+1, \psi U \vartheta) = 1]$

and the following global condition

$$\forall i \quad \beta(i, \psi U \vartheta) = 1 \; \rightarrow \; \exists k \geq i \; \beta(k, \vartheta) = 1$$

The local conditions are checked in the transitions of Δ. The global transition is checked by the acceptance sets F_ψ. $\square$

Theorem 3 (Landweber)**.** *Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, \mathcal{F})$ be a DMA.*

1. *$L(\mathcal{A})$ is DBA-recognizable iff $\mathcal{F}$ is closed under super loops.*
2. *$L(\mathcal{A})$ is E-recognizable iff $\mathcal{F}$ is closed under reachable loops.*

Proof. .

Claim : $L(\mathcal{A})$ is DBA-recognizable $\Rightarrow$ $\mathcal{F}$ is closed under super loops.

$L(\mathcal{A}) = \lim(K)$ for some regular K. Let $S \subseteq S'$ be two loops with $S \in \mathcal{F}$. Let $q \in S$ and $q_0 \overset{u}{\rightarrow} q \underset{S}{\overset{v}{\rightarrow}} q \underset{S'}{\overset{w}{\rightarrow}} q$. We can find $i_1, i_2, \ldots$ such that all words $uv^{i_1}wv^{i_2}w\ldots$ have prefixes in K. Hence, the union of these words is accepted by $\mathcal{A}$, so $S' \in \mathcal{F}$.

Claim : $\mathcal{F}$ is closed under super loops $\Rightarrow$ $L(\mathcal{A})$ is DBA-recognizable.

A DBA for the language uses states $Q \times 2^Q$. It uses the first component to simulare $\mathcal{A}$ and the second to collect visited states. As soon as the second component is in $\mathcal{F}$, it is reset to $\emptyset$. The final states are $Q \times \{\emptyset\}$.

Claim : $L(\mathcal{A})$ is E-recognizable $\Rightarrow$ $\mathcal{F}$ is closed under reachable loops.

$L(\mathcal{A}) = K \cdot \Sigma^\omega$ for some regular K. Let $q_0 \overset{u}{\rightarrow} q \underset{\in \mathcal{F}}{\overset{v}{\rightarrow}} q$. Then there must be a $w \sqsubset uv^\omega$ with $w \in K$ and $w \sqsubseteq uv^n$. Then any loop that is reachable from q must be accepting.

Claim : $\mathcal{F}$ is closed under reachable loops $\Rightarrow$ $L(\mathcal{A})$ is E-recognizable.

One can simply use the DMA as an E automaton with $F = \bigcup \mathcal{F}$. $\square$

Theorem 4. *NBAs, S1S, S1S$_0$, and $\exists$S1S have the same expressive power.*

Proof. **NBA ⇒ ∃S1S** Let $\mathcal{A} = (Q, \mathbb{B}^n, 1, \Delta, F)$ be the NBA with $Q = \{1, \ldots, m\}$. We encode a run of $\mathcal{A}$ on a word with a formula.

$$\varphi_{\mathcal{A}}(X_1, \ldots, X_n) = \exists Y_1 \ldots \exists Y_m \mathrm{Part}(\overline{Y}) \wedge Y_1(0) \wedge \mathrm{Trans}(\overline{X}, \overline{Y}) \wedge \mathrm{Fin}(\overline{Y})$$

$$\mathrm{Part}(Y_1, \ldots, Y_m) = \forall x \left(\bigvee_{i=1}^{m} Y_i x \wedge \bigwedge_{i=1}^{m} \bigwedge_{j \neq i} \neg Y_i x \vee \neg Y_j x \right)$$

$$\mathrm{Trans}(\overline{X}, \overline{Y}) = \forall x \bigvee_{\tau \in \Delta} \psi_\tau(x, \overline{X}, \overline{Y})$$

$$\psi_{(p,a,q)}(x, X_1, \ldots, X_n, Y_1, \ldots, Y_m) = Y_p x \wedge Y q(x+1) \wedge X_a x$$

$$\mathrm{Fin}(Y_1, \ldots, Y_m) = \forall x \exists y \left(x < y \wedge \bigvee_{q \in F} Y_q y \right)$$

∃S1S ⇒ S1S trivial

S1S ⇒ S1S$_0$ Let $\varphi \in$ S1S. We define an equivalent $\varphi' \in$ S1S$_0$ inductively.

- Boolean operators stay the same.
- $\exists x \psi(x) \mapsto \exists X_x(\mathrm{Sing}(X_x) \wedge \psi'(X_x))$
- $Xx \mapsto X_x \subseteq X$
- $x + 1 = y \mapsto \mathrm{Succ}(X_x, X_y)$

S1S$_0$ ⇒ NBA $\exists$, $\vee$, $\neg$ correspond to projection, union, and complement of automata respectively. Therefore it suffices to give NBAs for the three base formulas.

- $X_1 \subseteq X_2$: $Q = F = \{q_0\}$, $\Delta = \{(q_0, a, q_0) \mid a \neq (1\ 0)\}$.
- $\mathrm{Sing}(X)$: $Q = \{q_0, q_f\}$, $F = \{q_f\}$, $\Delta = \{(q_0, 0, q_0), (q_f, 0, q_f), (q_0, 1, q_f)\}$.
- $\mathrm{Succ}(X_1, X_2)$: $Q = \{q_0, q_1, q_2\}$, $F = \{q_2\}$,
  $\Delta = \{(q_0, (0\ 0), q_0), (q_2, (0\ 0), q_2), (q_0, (1\ 0), q_1), (q_1, (0\ 1), q_2)\}$.

The complexity of this construction is $2 \uparrow\uparrow |\varphi|$.

□

Definition 1. *Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ be an NBA and $\alpha \in \Sigma^\omega$. The run tree $T(\mathcal{A}, \alpha)$ is a tree defined as follows:*

- *Nodes are labelled by Q.*
- *The root is labelled q_0.*

- *For a node v with label q, let $P = \{p \in Q \mid (q, \alpha(|v|), p) \in \Delta\}$. Then v has a child labeled p for every $p \in P$.*

The left-right run tree $T_{LR}(\mathcal{A}, \alpha)$ is a binary tree defined as follows:

- *Nodes are labelled by 2^Q.*
- *The root is labelled $\{q_0\}$.*
- *For a node v with label P, let $P' = \{p' \in Q \mid \exists p \in P : (p, \alpha(|v|), p') \in \Delta\}$. Then v has a left child labelled $P \cap F$ and a right child labelled $P \setminus F$.*

The reduced LR run tree $R(\mathcal{A}, \alpha)$ is a binary tree defined as follows:

1. *Let $T = T_{LR}(\mathcal{A}, \alpha)$.*
2. *For all nodes v, v' in T with labels P, P' such that v occurs on the same level left of v': label v' by $P' \setminus P$.*
3. *Remove all nodes labeled $\emptyset$ from T.*

The marked reduced LR run tree $M(\mathcal{A}, \alpha)$ is a binary tree defiend as follows:

- *The structure is the same as $R(\mathcal{A}, \alpha)$, only the labels differ. Nodes are labeled by $2^Q \times Tok \times \{red, yellow, green\} \times \{1, \ldots, |Q|\}$ where $Tok = \{t_1, \ldots, t_{|Q|}\}$ and the last component corresponds to the age of the token.*
- *The root is labeled $\{t_1, yellow, 1\}$. Let $v_1, \ldots, v_k$ be nodes of layer n with labels $l_1, \ldots, l_k$ and let $s_1, \ldots, s_k$ be the sets of their successors. The labels of layer $n + 1$ are defined as follows:*
  1. *For $i \in \{1, \ldots, k\}$, let $s_1 \cup \cdots \cup s_i = \{u_1, \ldots, u_j\}$ in order from left to right. Let u be the right-most node of these successors. Move the token from l_i to u; color it yellow if $u \in s_i$ and green otherwise. If there are multiple tokens assigned to u, keep only the oldest one.*
  2. *Let T be the set of tokens that are placed on a node on level $n+1$ and let T' be the other tokens. Shift the ages of tokens in T so that they are all older than tokens in T'. Then move all tokens in T' to arbitrary empty nodes on this level and color them red.*

Theorem 5 (McNaughton)**.** *A language is NBA-recognizable iff it is DMA-recognizable.*

Proof. $\Leftarrow$ NBA with $L(\mathcal{A}) = \bigcup_{F \in \mathcal{F}} \left(\bigcap_{q \in F} L(\mathcal{A}_q) \cap \bigcap_{q \notin F} \overline{L(\mathcal{A}_q)} \right)$ where $\mathcal{A}_q$ is $\mathcal{A}$ starting in q.

$\Rightarrow$ Let $\mathcal{A}$ be an NBA and $\alpha \in \Sigma^\omega$. We prove the following claims:
Claim 1: In $M(\mathcal{A}, \alpha)$, there is a token that is green infinitely often and red only finitely often $\Leftrightarrow$ $R(\mathcal{A}, \alpha)$ has a path that branches left infinitely often.
Claim 2: $R(\mathcal{A}, \alpha)$ has a path that branches left infinitely often $\Leftrightarrow$ $\mathcal{A}$ accepts α.

If these claims are true, then a Muller automaton with state space $(|Q| + 1)^Q \times T^{\text{Tok}}$ where $T = (|Q|+1) \times |Q| \times \{r, g, y\}$ can check the correct construction from level to level in the tree. The condition for a correct token is easily formulated as a Rabin condition. The size of this automaton is $2^{\mathcal{O}(|Q| \times \log |Q|)}$.

Claim 1 Let $l \in \mathbb{N}$ be a depth in the tree at which point:

- all infinite paths have seperated
- all infinite paths that only branch left finitely often only branch right below l
- all tokens that are colored red only finitely often are only colored yellow or green below l

Let $v_1, \ldots, v_m$ be the nodes on level l in order from left to right.

$\Rightarrow$ Let $t \in \mathrm{Tok}$ be a token that becomes green infinitely often and red only finitely often and let v_i be the node that t is placed on. By choice of l, t never becomes red so there must be an infinite path π that starts at some v_j left of v_i ($j \leq i$). If we choose j maximal, then t reaches π at some point; because the infinite paths are all seperated, π is unique. π is an infinitely left branching path; otherwise it would only branch right and therefore, t would only be yellow from this point on.

$\Leftarrow$ Let v_i be a node at which a path π starts that branches left infinitely often. If there is another infinite path π' that starts right of π, let v_j be the node that it passes through; otherwise let $j := m+1$. Let t be the oldest token on $v_1, \ldots, v_{j-1}$. t never moves left of π until it is overwritten by an older token.

If $j = m + 1$ it is clear that an overwrite can never happen. Otherwise, all older tokens right of π are "blocked" by π' and therefore also never overwrite t. Hence, t never becomes red below l. It remains to be shown that it becomes green infinitely often. Note that t at some point must reach π since it is the only infinite path in the movement zone of t. π branches left infinitely often which proves the claim.

Claim 2 $\Rightarrow$ Let $\pi = Q_0Q_1\ldots$ be a path in $R(\mathcal{A}, \alpha)$ that branches left infinitely often. We construct a run $q_0q_1\ldots$ with $q_i \in Q_i$. That run must be accepting of $\mathcal{A}$ on α.

Consider the tree T constructed from $T(\mathcal{A}, \alpha)$ in which on every level i, all nodes $q \notin Q_i$ are removed. This tree is finitely branching (as $T(\mathcal{A}, \alpha)$ was already finitely branching) but still infinite because π is infinite. By König's lemma, there is an infinite path in T with the desired constraint.

$\Leftarrow$ Let ρ be an accepting run of $\mathcal{A}$ on α. Consider R' constructed from $R(\mathcal{A}, \alpha)$ in which on every level i, all nodes to the right of $\rho(i)$ are removed. The remaining nodes which contain ρ form an infinite run π. Since ρ is accepting, π branches left infinitely often. □

Theorem 6. *For each Muller automaton, one can construct an equivalent parity automaton. The construction preserves determinism.*

Proof. Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \mathcal{F})$ be a Muller automaton with $Q = \{q_0, \ldots, q_{n-1}\}$. We define the LAR automaton $\mathcal{A}_{\mathrm{LAR}} = (\mathrm{LAR}(Q), \Sigma, [q_0q_1 \ldots q_{n-1}, 1], \Delta_{\mathrm{LAR}}, c_{\mathrm{LAR}})$ as follows:

- $\mathrm{LAR}(Q)$ contains an ordering of Q and a so called hit-marker: $[p_1 \ldots p_n, h]$ with $1 \leq h \leq n$. This means that the most recently seen state was p_1 which was at position h in the list before.
- $c_{\mathrm{LAR}}([p_1 \ldots p_n, h]) := \begin{cases} 2h & \text{if } P \in \mathcal{F} \\ 2h-1 & \text{if } P \notin \mathcal{F} \end{cases}$ where $P = \{p_1, \ldots, p_h\}$.
- $\Delta_{\mathrm{LAR}} : [p_1 \ldots p_n, h] \xrightarrow{a} \mathrm{up}([p_1 \ldots p_n, h], p')$ for all $(p_1, a, p') \in \Delta$, where $\mathrm{up}([p_1 \ldots p_n, h], p') = [p'p_1 \ldots p_{i-1}p_{i+1} \ldots p_n, i]$.

Claim : Let $\rho \in Q^\omega$ be a run of $\mathcal{A}$ on some word α and let $\rho' \in (\mathrm{LAR}(Q))^\omega$ be the corresponding run of $\mathcal{A}_{\mathrm{LAR}}$. Then in ρ', the hit marker h is greater than $|\mathrm{Inf}(\rho)|$ only finitely often; and the hit segment $\{p_1, \ldots, p_h\}$ equals $\mathrm{Inf}(\rho)$ infinitely often. □

Theorem 7. *For each ABA, one can construct an equivalent NBA with states at most* $3^{|Q|}$.

Proof. Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, F)$ be an ABA. We define $\mathcal{A}' = (\{0,1,2\}^Q, \Sigma, f_0, \Delta, \{0,2\}^Q)$ with $f_0(q) = \begin{cases} 1 & \text{if } q = q_0 \\ 0 & \text{else} \end{cases}$ and Δ as described below.

A state is a function $f : Q \to \{0,1,2\}$. Consider a run-tree of $\mathcal{A}$ on some word and all states that are "active" on one level in the tree. $f(q)$ is 0 for all states which are not active and 1 or 2 for all others. A state is assigned to 1 by default, which just means that it is active in some path. It is assigned 2 if "recently" on all paths it is active on, a final state in F was seen. Recently means that it occurred since the last time that an entire level in the run tree was made up of final states.

Formally, we have $(f, a, g) \in \Delta$ for all $g : Q \to \{0,1,2\}$ which satisfy the following:

- The active states need to be passed on, i.e. for all $q \in Q$: if $f(q) \in \{1,2\}$ then there must be an $X_q \subseteq Q$ with $X_q \models \delta(q,a)$ such that $g(X_q) \subseteq \{1,2\}$.
- A state is assigned 2 if it is final, i.e. $g(q) = 2$ if $q \in F$.
- Otherwise, a 1 overwrites a 2 for succession, i.e. for all $p \in P$, if there is a q with $f(q) = 1$ and $p \in X_q$, then also $g(p) = 1$.
- If all states are marked with a 2, $\mathcal{A}'$ reached a final state. We reset the values to $g(p) = 2$ iff $p \in F$.

□

2 Tree Automata

Theorem 8. *Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ be a TWA. There is an NTA $\mathcal{A}'$ of size exponential in $|Q|$ that recognizes $T(\mathcal{A})$.*

Proof. Let $\sim_{T(\mathcal{A})}$ be the usual equivalence relation, i.e. $t_1 \sim_{T(\mathcal{A})} t_2$ iff $\forall s \in S_\Sigma : s \cdot t_1 \in T(\mathcal{A}) \leftrightarrow s \cdot t_2 \in T(\mathcal{A})$. We define a relation $\sim \subseteq T_\Sigma \times T_\Sigma$ such that $\text{index}(\sim_{T(\mathcal{A})}) \leq \text{index}(\sim) \leq 2^{|Q|^2 \cdot m+1}$, where m is the maximal rank in Σ.

Let $t_0 \in T_\Sigma$ and $a_m \in \Sigma_m$ be arbitrary. For every $t \in T_\Sigma$ and $1 \leq i \leq m$, we define $t^{(i)} = a_m(t_0, t_0, \ldots, t, \ldots, t_0)$, meaning the i-th subtree below the root is t. Further, we define a relation $B_t^i \subseteq Q \times Q$ with $(p, q) \in B_t^i$ iff there is a run segment ρ of $\mathcal{A}$ on $t^{(i)}$, such that the run begins at the root of t, never leaves that subtree until the end. Meaning, $\rho = (p, i)(q_1, iu_1) \ldots (q_n, iu_n)(q, \varepsilon)$.

Finally, let $t_1 \sim t_2$ iff $t_1 \in T(\mathcal{A}) \leftrightarrow t_2 \in T(\mathcal{A})$ and $\forall i : B_{t_1}^i = B_{t_2}^i$.

Idea: $(p, q) \in B_t^i$ if $\mathcal{A}$ can enter t as i-th child with state p and after some while leaves it again with state q.

Claim : Let $t_1 \sim t_2$. Then $t_1 \sim_{T(\mathcal{A})} t_2$.

Let $s \in S_\Sigma$. Due to the symmetric definition of $\sim$, it suffices to show that $t_1 \in T(\mathcal{A})$ implies $t_2 \in T(\mathcal{A})$, so let $t_1 \in T(\mathcal{A})$. If $s = \circ$, then $s \cdot t_1 = t_1 \in T(\mathcal{A})$. By definition of $\sim$, this implies $s \cdot t_2 = t_2 \in T(\mathcal{A})$.

Otherwise $s \neq \circ$. Let $\rho_1\rho_2\rho_3$ be an accepting run of $\mathcal{A}$ on $s \cdot t_1$ such that ρ_1 only stays outside of t_1 and ρ_2 only stays inside of t_1. Since $B_{t_1}^i = B_{t_2}^i$, there is a run segment of $\mathcal{A}$ on t_2 which enters and exits the tree with the same states as ρ_2 does, meaning it can replace ρ_2 in the accepting run. Repeating this procedure gives an accepting run of $\mathcal{A}$ on $s \cdot t_2$, so $t_2 \in T(\mathcal{A})$.

Notes on the construction: each state in the NTA corresponds to a list of Q-states that $\mathcal{A}$ had when visiting this node, together with the direction from which it was coming. That can be used to check the correctness of a run. □

Theorem 9. *A language of finite trees $T \subseteq T_\Sigma$ can be recognized by an NTA iff it can be described by a regular tree expression.*

Proof. $\Rightarrow$ Let $\mathcal{A} = (Q, \Sigma, \Delta, F)$ be an NTA for T. For $R, I \subseteq Q, q \in Q$, we define $T(R, I, q) \subseteq T_{\Sigma \cup C_R}$ (where $C_R = \{c_p \mid p \in R\}$) as the set of all trees on which $\mathcal{A}$ has a run that only uses states in I and ends in q. We can inductively define regular expressions for $T(R, I, q)$. Then $T(\mathcal{A}) = \bigcup_{q \in F} T(\emptyset, Q, q)$.

For all $R \subseteq Q, q \in Q$, $T(R, \emptyset, q)$ contains only trees of height 0 or 1, so it is a finite union of singular languages.

$$T(R, I \cup \{i\}, q) = T(R, I, q) + T(R \cup \{i\}, I, q) \cdot^{c_i} (T(R \cup \{i\}, I, i))^{*_{c_i}} \cdot^{c_i} T(R, I, i)$$

$\Leftarrow$ Show by induction that a regular expression r can be transformed to a NTA $\mathcal{A}_r$.

- If $r = t \in T_{\Sigma\cup C}$, then there is an automaton $\mathcal{A}_t$ with $T(\mathcal{A}_t) = \{t\}$.
- If $r = s + t$, then $\mathcal{A}_r$ is the union automaton of $\mathcal{A}_s$ and $\mathcal{A}_\sqcup$.
- If $r = s \cdot^c t$, then $\mathcal{A}_r = (Q_s \cup Q_t, \Sigma \cup C, \Delta_r, F_s)$, where Δ_r simulates $\mathcal{A}_s$ and $\mathcal{A}_t$ but the final transitions in $\mathcal{A}_t$ are replaced by initial transitions to $\mathcal{A}_s$.
- If $r = s^{+_c}$, let $\mathcal{A}_r = (Q_s, \Sigma \cup C, \Delta_r, F_s)$, where Δ_r simulates $\mathcal{A}_s$ and allows "restarts" when a final state could be reached.

□

Theorem 10. *Let D be an EDTD. We can construct a NUTA $\mathcal{A}$ with $T(D)^{\complement} = T(\mathcal{A})$ in polynomial time in $|D|$.*

Proof. Let $D = (\Sigma', P, S^{(1)})$. For $a^{(i)} \in \Sigma'$, let $a^{(i)} \to r_{a,i}$ be the according rule in P, where $r_{a,i}$ is a regular expression that can be transformed to a DFA in polynomial time. If $b \in \Sigma$ occurs in $r_{a,i}$ then it must have a unique type j (since D is single-typed). We call $b^{(j)}$ the b-successor of $a^{(i)}$. If b does not occur in $r_{a,i}$, we say that the b-successor is $b^{\perp}$. Furthermore, we assume $r_{a,\perp} = \varepsilon$ for all a.

We define a typing function $f : \mathrm{dom}_t \to \{\perp, 1, \ldots, k\}$. We assign $f(\varepsilon) = \begin{cases} 1 & \text{if } \mathrm{val}_t(\varepsilon) = S \\ \perp & \text{else} \end{cases}$.
For the other nodes, let u be a node with parent v. We call $\mathrm{val}_t(v) = a$ and $\mathrm{val}_t(u) = b$. Then there is a unique $i \in \{\perp, 1, \ldots, k\}$ such that $b^{(i)}$ is the b-successor of $a^{f(v)}$. We set $f(u) := i$.

Claim: $t \in T(D)$ iff $\forall v \in \mathrm{dom}_t : f(v) \neq \perp$ and $a_1^{f(v1)} \ldots a_m^{f(vm)} \in L(r_{a,f(v)})$, where $a = \mathrm{val}_t(v)$ and $a_j = \mathrm{val}_t(vj)$. (without proof)

Using this claim, we can provide an automaton $\mathcal{A} = (Q, \Sigma, \Delta, F)$ that accepts a tree iff it violates the constraint.

- $Q = \Sigma \times \{\perp, 1, \ldots, k\} \times \{0, 1\}$, where the last component denotes whether a violation was found.
- $\Delta = \{(L_{a,i,x}, a, (a, i, x)) \mid a \in \Sigma, i \in \{\perp, 1, \ldots, k\}, x \in \{0, 1\}\}$
  Let $\alpha = (a_1, i_1, x_1) \ldots (a_m, i_m, x_m) \in Q^m$ and $w := a_1^{(i_1)} \ldots a_m^{(i_m)} \in (\Sigma')^m$.
  - succ :⇔ for all $1 \leq j \leq m$, $a_j^{(i_j)}$ is the a_j-successor of $a^{(i)}$.
  - sat_0 :⇔ $w \in L(r_{a,i})$ and for all $1 \leq j \leq m$, $x_j = 0$ and $i_j \neq \perp$.
  - sat_1 :⇔ $w \notin L(r_{a,i})$ or there is a $1 \leq j \leq m$ such that $x_j = 1$ or $i_j = \perp$.

  Then $\alpha \in L_{a,i,x}$ iff succ and sat_x hold.
- $F = \{(S, 1, 1)\} \cup \{(a, i, x) \mid a \neq S\}$, meaning either a violation was found or the starting symbol was not S.

□

Theorem 11. *The class of DTWA-recognizable languages is closed under complement.*

Proof. Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, \{q_f\})$ be a DTWA that only moves to q_f at the root. The **backwards configuration graph** $\mathrm{BCG}(\mathcal{A}, t)$ is defined as a tree over $Q \times \mathrm{dom}_t$ with root (q_f, ε). For a node (q, u), the children are all (p, v) such that $(p, v) \to_{\mathcal{A}} (q, u)$. We define $\overline{\mathcal{A}}$ in a way that it performs DFS on the BCG of the input tree and accepts iff the node (q_0, ε) is found.

For that, let $\prec \subseteq (Q \times \mathrm{Dir})^2$ be an arbitrary linear order on $Q \times \mathrm{Dir}$. We set $\overline{\mathcal{A}} = (\overline{Q}, \Sigma, q_0, \overline{\delta}, q_f)$ with $\overline{Q} = \{q_0, q_f\} \cup \{\langle p, (q, d)\rangle \mid p, q \in Q, d \in \mathrm{Dir}\}$. The behavior of $\overline{\delta}$ is described below. Let $\langle q, (q', d)\rangle$ be a state.

Case 1 : In the ordering $\prec$, $(\hat{q}, \hat{d})$ is the next largest element after (q', d). (for q_0 we also consider this case with the $\prec$-minimal pair.)

Case 2 : (q', d) $\square$

Theorem 12. *Let $T \subseteq T_\Sigma$. T is regular iff fcns(T) is regular.*

Proof. $\Rightarrow$ Let $\mathcal{A} = (Q, \Sigma, \Delta, F)$ be a NUTA with $T(\mathcal{A}) = T$. Wlog we assume that $\mathcal{A}$ is normalized. For every transition $\tau = (L_{a,q}, a, q) \in \Delta$, let $\mathcal{B}_{a,q} = (P_{a,q}, Q, p^0_{a,q}, \Delta_{a,q}, F_{a,q})$ be a NFA with $L(\mathcal{B}_{a,q}) = L_{a,q}$. We define $\mathcal{A}_{\mathrm{fcns}} = (Q_{\mathrm{fcns}}, \Gamma, \Delta_{\mathrm{fcns}}, F_{\mathrm{fcns}})$ so that $T(\mathcal{A}_{\mathrm{fcns}}) = \mathrm{fcns}(T)$.

- $Q_{\mathrm{fcns}} = \{q_f, q_\#\} \cup \bigcup_{a \in \Sigma, q \in Q} P_{a,q}$
- $F_{\mathrm{fcns}} = \{q_f\}$
- Δ_{fcns}:
  - $(\#, q_\#)$
  - For all $p \in \bigcup_{a \in \Sigma, q \in Q} F_{a,q}$: $(\#, p)$
  - For all $a \in \Sigma, q \in F$: $(p^0_{a,q}, q_\#, a, q_f)$
  - For all $a \in \Sigma, p \in P, p' \in P, q \in Q$ with $(p, q', p') \in \bigcup_{b \in \Sigma, q' \in Q} \Delta_{b,q'}$: $(p^0_{a,q}, p', b, p)$

Via induction on $t_1 \ldots t_n$, one can show that $p \in \Delta^*_{\mathrm{fcns}}(\mathrm{fcns}(t_1 \ldots t_n))$ iff there are $q_1, \ldots q_n \in Q$ such that $\forall i : q_i \in \Delta^*(t_i)$ and $q_1 \ldots q_n \in L(\mathcal{B}_{a,q}$ init $p)$.

$\Leftarrow$ Let $\mathcal{A} = (Q, \Sigma, \Delta, F)$ be an NTA with $T(\mathcal{A}) = \mathrm{fcns}(T)$. We define $\square$

3 Infinite Games (Trees)

Definition 2. *Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, Acc)$ be a tree automaton and $t : \mathbb{B}^* \to \Sigma$. The membership game $\mathcal{G}_{\mathcal{A},t}$ checks whether t is accepted by $\mathcal{A}$.*

Starting from (ε, q_0), player 0 (called Automaton) chooses for (u, q) a transition $(q, t(u), q_1, q_2) \in \Delta$. Then player 1 (called Pathfinder) chooses either $i = 0$ or $i = 1$ and moves to position (ui, q_i). A play $(\varepsilon, q_0)(\varepsilon, \tau_0)(u_1, q_1)(u_1, \tau_1)\dots$ is accepting iff $q_0 q_1 \dots \in Acc$.

We assume that Acc is a parity condition in general.

Theorem 13. *$t \in T(\mathcal{A})$ iff Automaton wins $\mathcal{G}_{\mathcal{A},t}$*

Proof. If $\mathcal{A}$ is PTA, then $\mathcal{G}_{\mathcal{A},t}$ is a parity game and therefore positionally determined, i.e. winning strategies for player Automaton can be seen as $\sigma : \mathbb{B}^* \times Q \to \Delta$. Winning strategies can be transformed to accepting runs and vice-versa. □

Theorem 14. *PTA-recognizable tree languages are closed under complement.*

Proof. Let $\mathcal{A}$ be a PTA. For a tree t, we consider the strategy extension $t^\frown s$ where s is a tree over $\Gamma = \mathbb{B}^\Delta$ which corresponds to a strategy of Pathfinder. The node is implicitly given in the tree and the state will be captured by an automaton. This gives a tree over $\Sigma \times \Gamma$.

Now consider words over $\Sigma \times \Gamma \times \mathbb{B}$ which describe a path in such a tree $t^\frown s$.

Step 1 : Construct a parity automaton $\mathcal{B}$ that checks whether a word over $\Sigma \times \Gamma \times \mathbb{B}$ represents a valid path in such a tree that can be won by player Automaton against the given Pathfinder strategy.

$\mathcal{B}$ uses state set Q. From a state $q \in Q$ with symbol (a, h, i), the automaton guesses a transition $\tau = (q, a, q_1, q_2)$ with $h(\tau) = i$ and moves to q_i.

Step 2 : Construct a deterministic parity automaton $\mathcal{A}^{\text{path}}_{\text{strat}}$ that checks whether a word over $\Sigma \times \Gamma \times \mathbb{B}$ represents a valid path in a tree that is won by player Pathfinder with the given strategy.

This can be done by determinizing and complementing $\mathcal{B}$.

Step 3 : Construct a deterministic PTA $\mathcal{A}_{\text{strat}}$ that checks whether the given strategy is winning for Pathfinder in $\mathcal{G}_{\mathcal{A},t}$.

Define $\delta_{\text{strat}}(q, (a, h)) = (\delta^{\text{path}}_{\text{strat}}(q, (a, h, 0)), \delta^{\text{path}}_{\text{strat}}(q, (a, h, 1)))$.

Step 4 : Construct a PTA $\mathcal{A}^{\complement}$ that checks whether there exists a winning strategy for Pathfinder in $\mathcal{G}_{\mathcal{A},t}$.

Use projection to define $(q, a, q_1, q_2) \in \Delta$ iff there is an $h \in \Gamma$ with $\delta_{\text{strat}}(q, (a, h)) = (q_1, q_2)$. □