

1 Infinite Words

Theorem 1.1. *Every non-empty ω -regular language contains an ultimately periodic word.*

Theorem 1.2. *For a Kripke structure \mathcal{K} with initial state s and $\varphi \in \text{LTL}$, the model checking problem $L(\mathcal{K}, s) \subseteq L(\varphi)?$ is PSPACE-complete.*

Proof. **PSPACE** Compute the intersection automaton for $L(\mathcal{K}, s) \cap L(\neg\varphi)$ and test it for emptiness.

PSPACE-hard Encode a poly.-length Turing tape as a Kripke structure and its correct behavior in LTL. \square

Theorem 1.3 (Büchi). *The MSO theory of $(\mathbb{N}, +1, <, 0)$ is decidable.*

Proof. Corresponds to S1S formula. Can be checked with NBA emptiness test. \square

Theorem 1.4. *The FO theory of $(\mathbb{R}, +, <, 0)$ is decidable.*

Proof. Encode real numbers x by triples of sets (X_s, X_i, X_f) with the number's sign ($X_s = \emptyset$ or $\{0\}$), the positive decimal digits in binary encoding, and the positive fractional digits in binary encoding. Then an FO sentence can be transformed to an equi-satisfiable MSO sentence over $(\mathbb{N}, +1, <, 0)$. \square

Theorem 1.5. *Subset-construction does not suffice to determinize NBAs.*

Theorem 1.6. *For every n , there is $L_n \subseteq \Sigma^\omega$ s.t. there is an NBA that recognizes L_n with $n + 2$ states, but every det. Rabin automaton that recognizes L_n has at least $n!$ states.*

Theorem 1.7. *There is a DBA-recog. language which does not have a unique minimal DBA. DBAs minimized with the DFA minimization algorithm can be arbitrarily bad compared to a minimal DBA.*

Theorem 1.8. *Weak DBAs can be minimized uniquely in polynomial time.*

Theorem 1.9. *Given an ABA \mathcal{A} , the dual $\tilde{\mathcal{A}}$ is an alternating co-Büchi automaton which accepts $\overline{L(\mathcal{A})}$, with $\tilde{F} = Q \setminus F$ and $\tilde{\delta}$ exchanging true/false and \wedge/\vee .*

1.1 Simulation Game

Definition 1. *Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ be an NBA. We define the delayed simulation game $\mathcal{G}_{\mathcal{A}}(G_{\mathcal{A}}, \text{Win})$ as follows*

- $G_{\mathcal{A}} = (V_0, V_1, E, c)$
- $V_0 = \Sigma \times Q \times Q$
- $V_1 = Q \times Q$
- *Player 0 moves from (a, p, q) to (p, p') with $(q, a, p') \in \Delta$*
- *Player 1 moves from (q, q') to (a, p, q') with $(q, a, p) \in \Delta$*

- $c : V \rightarrow \{-1, 0, 1\}$ with $c(v) = \begin{cases} -1 & \text{if } v \in F \times (Q \setminus F) \\ 1 & \text{if } v \in Q \times F \\ 0 & \text{otherwise} \end{cases}$

- $\alpha \in \text{Win}$ iff after every -1 in α , there is a 1 later on.

Write $q \preceq_{de} q'$ if player 0 has a winning strategy from (q, q') .

Idea: Player 1 chooses symbols in Σ and transitions on the second state. Player 0 has to answer with transition on the first state which lead to a run of the same acceptance.

Theorem 1.10. *If $q \preceq_{de} q'$ and $q' \preceq_{de} q$, then q and q' can be merged in \mathcal{A} without changing the language of the automaton.*

Theorem 1.11. *The delayed simulation game can be reduced to a Büchi game in linear time.*

2 Finite Trees

Theorem 2.1 (Pumping Principle). *Let $T \subseteq T_\Sigma$ be a regular ranked tree language. There is a $n \in \mathbb{N}$ such that for all trees $t \in T$, all $m > n$, and all paths $\pi_1 \dots \pi_m$, there are $1 \leq i < j \leq m$ such that for all $k \in \mathbb{N}$:*

$$t[\circ/u] \cdot (t[\circ/v]|_u)^k \cdot t|_v \in T$$

where $u = \pi_1 \dots \pi_i$ and $v = \pi_1 \dots \pi_j$.

Definition 2. Let $T \subseteq T_\Sigma$. The **Myhill-Nerode equivalence** is $\sim_T \subseteq T_\Sigma \times T_\Sigma$ with

$$t_1 \sim_T t_2 \Leftrightarrow \forall s \in S_\Sigma : s \cdot t_1 \in T \Leftrightarrow s \cdot t_2 \in T$$

The index of T is $\text{Index}(\sim_T) := |T / \sim_T|$.

Theorem 2.2. *Let $T \subseteq T_\Sigma$. T is regular iff $\text{Index}(\sim_T)$ is finite. If T is regular, \mathcal{A}_T is the minimal DTA.*

Theorem 2.3. *The emptiness problem for NTAs can be reduced to HORN-SAT in linear time.*

2.1 BTTs

Theorem 2.4. *The equivalence problem for BTTs is undecidable.*

Proof. Follows from the equivalence problem of rational word transducer. □

Theorem 2.5. *The emptiness problem for BTTs is decidable in polynomial time.*

Proof. One can construct in polynomial time a \uparrow NTA that recognizes the domain of a given BTT. The emptiness problem of BTTs then is equivalent to the emptiness problem of \uparrow NTAs. □

Theorem 2.6. *The type-checking problem (given regular T, T' , is $\mathcal{A}(T) \subseteq T'$?) is decidable.*

Proof. $\mathcal{A}(T) \subseteq T'$ iff $\mathcal{A}^{-1}((T')^c) \cap T = \emptyset$. If T' is regular then so is $\mathcal{A}^{-1}((T')^c) \cap T$ and this can be decided. □

Theorem 2.7. *If T is regular, then $\mathcal{A}^{-1}(T)$ is regular.
If \mathcal{A} is linear, then $\mathcal{A}(T_\Sigma)$ is regular.*

Theorem 2.8. *There are BTT-definable relations R_1, R_2 such that $R_1 \circ R_2$ is not BTT-definable.*

Proof. Let $\Sigma = \{f, g, h, c\}$ and for both $i \in \{1, 2\}$: $\mathcal{A}_i = (\{q, q_f\}, \Sigma, \Sigma, \Delta_i, \{q_f\})$ where

- Δ_1 :
 - $c \rightarrow q(c)$
 - $g(q(x_1)) \rightarrow q(g(x_1))$
 - $q(x_1) \rightarrow q_f(f(x_1))$
- Δ_2 :
 - $c \rightarrow q(c)$
 - $g(q(x_1)) \rightarrow q(g(x_1))$
 - $g(q(x_1)) \rightarrow q(h(x_1))$
 - $f(q(x_1), q(x_2)) \rightarrow q_f(f(x_1, x_2))$

\mathcal{A}_1 converts a tree $t \in T_{\{c, g\}}$ to $f(t, t)$. \mathcal{A}_2 replaces an arbitrary number of g s by h s. $R_1 \circ R_2$ defines the set of trees $f(t_1, t_2)$ where t_1 and t_2 are of equal height. \square

Theorem 2.9. *If \mathcal{A}_1 is linear **or** \mathcal{A}_2 is deterministic Then $R(\mathcal{A}_1) \circ R(\mathcal{A}_2)$ is BTT-definable.*

3 Infinite Trees

Theorem 3.1 (BTA Pumping). *For $t \in T_\Sigma, x \in \{0, 1\}^*, y \in \{0, 1\}^+,$ let*

$$t_{[x, y]}^* : \{0, 1\}^* \rightarrow \Sigma, z \mapsto \begin{cases} t(z) & \text{if } xy \not\sqsubseteq z \\ xz' & \text{if } \exists n > 0 : z = xy^n z' \text{ with } y \not\sqsubseteq z' \end{cases}.$$

Let \mathcal{A} be a BTA, $t \in T(\mathcal{A})$, ρ an accepting run of \mathcal{A} on t , and $x, y, y' \in \{0, 1\}^$ s.t. $\rho(x) = \rho(xy)$, $y' \sqsubset y$, and $\rho(xy') \in F$. Then $t_{[x, y]}^* \in T(\mathcal{A})$.*

Theorem 3.2. *Every non-empty regular tree language contains a regular tree.*

Theorem 3.3 (Rabin's Tree Theorem). *The MSO theory of T_2 is decidable for formulas $\varphi(X_1, \dots, X_n)$ and a model $X_1, \dots, X_n \subseteq \{0, 1\}^*$ is computable.*

Proof. Transform φ into an equivalent PTA. A model can be found by solving the emptiness game. \square