Rabin's Complementation
For every TAF over infinite trees, one can effectively construct an automaton \( \overline{A} \) that accepts the complement of the language accepted by \( A \)

1. Games / winning strategies
2. Partial acceptance condition
3. Determinacy \( \Rightarrow \) complementation
4. Finite games \( \Rightarrow \) decidability of nonemptiness
2 players

Automaton / Pathfinder

Position for automaton

Ant chooses a pair

\((q_1, q_2) \in S(q, a)\)

Position for Pathfinder chooses direction

0 or 1

Ant wins if for each infinite branch, the set of states seen along this branch satisfies Acc
Pantsy acceptance

\[ \exists i \quad \text{and} \quad 1_{\inf}(p) \cap \gamma_i = \emptyset \]

\[ \text{Inf}(p) \cap \gamma_i \neq \emptyset \quad \forall i \]

\[ \gamma \]

\[ \min_{\text{rank}} \inf(p) \]

\[ O_1 \subset E_1 \subset O_2 \subset E_2 \subset C \]
Pancy TA accept all regular tree languages
\[ p : q_0 \rightarrow q_f . \in Q^w \quad \text{Inf}(p) = F \]

1 2 3 4 - states

3 1 4 2 4 4 2 4 4

2 3 4 1 2 4 1 3 2 4 1 3 2 4 1 3 2 4

LAR

For each \( i < |Q| \), if the set of the last \( i \) elements of \( F \) odd rank
otherwise \( \rightarrow \) even rank

\[ F = 4 7, 13 \]
Suppose that $T$ is not accepted by $A$. Then Path has a memoryless winning strategy on $T$.

A memoryless strategy $f$ is:

$$f(s, \text{position}) \rightarrow 0, 1$$

Se $\gamma_1 \gamma_2 \in \Sigma^*$ and $\gamma \in Q \times \Sigma \times Q \times Q \rightarrow \gamma_1$ a tree labeled by "local instructions".
To check if Path has a winning strategy, guess $T_f$.

If

$\text{guess local inst } g$

For each in-f branch,

- For each assignment of states $q$ along this branch, local instructions applied to these states violate Acc
Finite games

\[ c \cup \mathcal{V}_1 \cup \mathcal{V}_2 \rightarrow C \]

Play leads to \( \omega \mathcal{C} \)

\( \text{Acc} \) over \( \omega \mathcal{C} \)

A) \( \text{Acc} \) is a parity condition.
Then it is decidable whether player 1 wins.

B) Consider input-free automata.
Winning = accepting some tree.
W. Thomas 'Logic Languages Automata' Handbook of Formal Languages LNCS 2500