Bernd Finkbeiner
Sven Schewe

Date: July 10, 2008

Automata, Games and Verification: Lecture 11

16 Parity Games

Assumptions:

- arena is finite or countably infinite.
- the number of colors is finite (max color k).

Lemma 1 (Merging strategies) *Given a parity game $\mathcal{G}$ and a set of nodes $U \subseteq V$, s.t. for every $p \in U$, Player σ has a memoryless strategy $f_{\sigma,p}$ that wins from p, then there is a memoryless winning strategy f_σ that wins from all $p \in U$.*

Proof:

- Index the positions in $V = \{p_0, p_1, p_2, \ldots\}$
- For $p_i \in V$, let $F_i \subseteq V$ be the set of positions that are reachable from p_i in plays that conform to f_{p_i}.
- Define $f_\sigma(q) = f_{\sigma,p_i}(q)$ for the smallest i such that $q \in F_i$.
- f is winning for Player 0:
  - Applying f_σ corresponds to applying f_{σ,p_i} with weakly decreasing i.
  - From some point onward, $i = i^*$ is constant.
  - The play is won because $f_{\sigma,p_{i^*}}$ is winning.

■

Theorem 1 *Parity games are memoryless determined.*

Proof:

Induction on k:

- $k = 0$: $W_0 = V, W_1 = \emptyset$. Memoryless winning strategy: fix arbitrary order on V. $f_0(p) = \min\{q \mid (p,q) \in E\}$.
- $k + 1$:
  - If $k + 1$, consider player $\sigma = 0$, otherwise $\sigma = 1$.
  - Let $W_{1-\sigma}$ be the set of positions where Player $(1-\sigma)$ has a memoryless winning strategy. We show that Player σ has a memoryless winning strategy from $V \setminus W_{1-\sigma}$.
  - Consider subgame $\mathcal{G}'$:

    * $V_0' = V_0 \setminus W_{1-\sigma}$;
    * $V_1' = V_1 \setminus W_{1-\sigma}$;
    * $E' = W \cap (V' \times V')$;
    * $c'(p) = c(p)$ for all $p \in V'$.
- $\mathcal{G}'$ is still a game:
    * for $p \in V_\sigma'$, there is a $q \in V \setminus W_{1-\sigma}$ with $(p,q) \in E'$, otherwise $p \in W_{1-\sigma}$;
    * for $p \in V_{1-\sigma}'$, for all $q \in V$ with $(p,q) \in E$, $q \in V \setminus W_{1-\sigma}$, hence there is a $q \in V'$ with $(p,q) \in E$.
- Let $C_i' = \{p \in V' \mid c'(p) = i\}$.
- Let $Y = \mathit{Attr}_\sigma'(C_{k+1}')$. ($\mathit{Attr}'$: Attractor set on $\mathcal{G}'$)
- Let f_A be the attractor strategy on $\mathcal{G}'$ into C_{k+1}'.
- Consider subgame $\mathcal{G}''$:
    * $V_0'' = V_0' \setminus Y$;
    * $V_1'' = V_1 \setminus Y$;
    * $E' = W \cap (V'' \times V'')$;
    * $C'' : V'' \to \{0, \ldots, k\}; c''(p) = c'(p)$ for all $p \in V''$.
- $\mathcal{G}''$ is still a game.
- Induction hypothesis: $\mathcal{G}''$ is memoryless determined.
- Also: $W_{1-\sigma}'' = \emptyset$ (because $W_{1-\sigma}'' \subseteq W_{1-\sigma}$: assume Player $(1-\sigma)$ had a winning strategy from some position in V''. Then this strategy would win in $\mathcal{G}$, too, since Player σ has no chance to leave $\mathcal{G}''$ other than to $W_{1-\sigma}$.)
- Hence, there is a winning memoryless winning strategy f_{IH} for player σ from V''.
- We define:

$$f_\sigma(p) = \begin{cases} f_{IH}(p) & \text{if } p \in V''; \\ f_A(p) & \text{if } p \in Y \setminus C_{k+1}'; \\ \text{min. successor in } V \setminus W_{1-\sigma} & \text{if } p \in Y \cap C_{k+1}'; \\ \text{min. successor in } V & \text{otherwise.} \end{cases}$$

- f_σ is winning for Player σ on $V \setminus W_{1-\sigma}$.
  Consider a play that conforms to f_σ:
    * Case 1: Y is visited infinitely often.
      $\Rightarrow$ Player σ wins (inf. often even color $k+1$).
    * Case 2: Eventually only positions in V'' are visited.
      $\Rightarrow$ Since Player σ follows f_{IH}, Player σ wins.

■

17 Tree Automata

Binary Tree: $T = \{0,1\}^*$.
Notation: T_Σ : set of all binary Σ-trees

Definition 1 *A tree automaton (over binary Σ-trees) is a tuple $\mathcal{A} = (S, s_0, M, \varphi)$:*

- *S: finite set of states*
- $s_0 \in S$
- $M = S \times \Sigma \times S \times S$
- *φ: acceptance condition (Büchi, parity, ...)*

Definition 2 *A run of a tree automaton $\mathcal{A}$ on a Σ-tree v is a S-tree (T, r), s.t.*

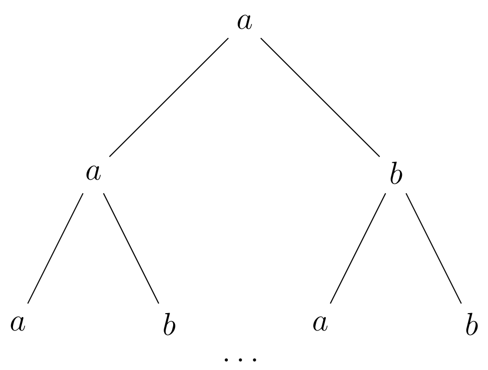
- $r(\epsilon) = s_0$
- $(r(q), v(q), r(q0), r(q1)) \in M$ *for all* $q \in \{0,1\}^*$

Definition 3 *A run is accepting if every branch is accepting (by φ). A Σ-tree is accepted if there exists an accepting run.*
$\mathcal{L}(A) :=$ *set of accepted Σ-trees.*

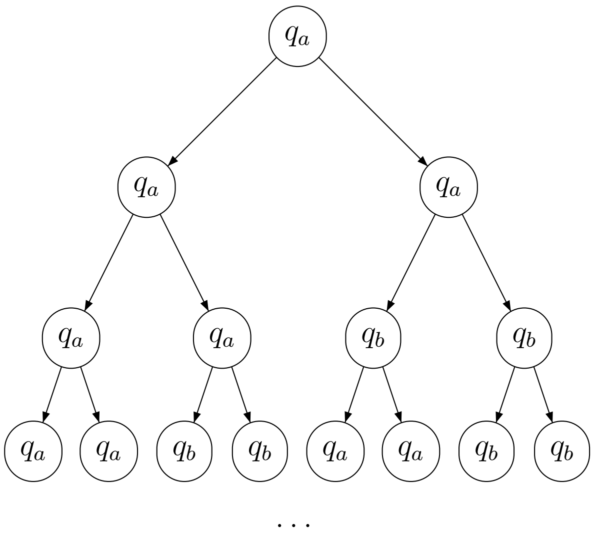
Example: $\{a, b\}$-trees with infinitely many bs on each path.

$\mathcal{A} = (S, s_0, M, c); \Sigma = \{a, b\}$;
$S = \{q_a, q_b\}; s_0 = q_a$;
$M = \{(q_a, a, q_a, q_a), (q_b, a, q_a, q_a), (q_a, b, q_b, q_b), (q_a, a, q_b, q_b), \ldots\}$;
Büchi $F = \{q_b\}$.

Σ-tree:



run:

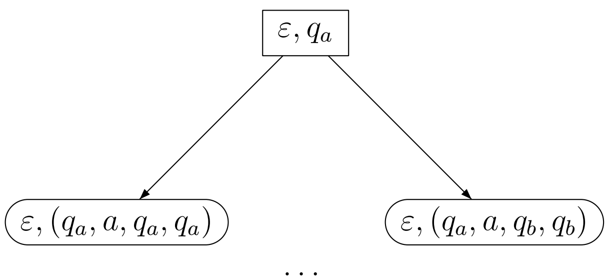


◆

Theorem 2 *A parity tree automaton* $\mathcal{A} = (S, s_0, M, c)$ *accepts an input tree* t *iff Player 0 wins the parity game* $\mathcal{G}_{\mathcal{A},t} = (V_0, V_1, E, c')$ *from position* (ε, s_0)*.*

- $V_0 = \{(w, q) \mid w \in \{0,1\}^*, q \in S\}$;
- $V_1 = \{(w, \tau) \mid w \in \{0,1\}^*, \tau \in M\}$;
- $E = \{((w, q), (w, \tau)) \mid \tau = (q, t(w), q'_0, q'_1), \tau \in M\}$
  $\cup \{((w, \tau), (w', q')) \mid \tau = (q, \sigma, q'_0, q'_1)$ *and*
  $((w' = w0$ *and* $q' = q'_0)$ *or* $(w' = w1$ *and* $q' = q'_1))\}$;
- $c'(w, q) = c(q)$ *if* $q \in S$;
- $c'(w, \tau) = 0$ *if* $\tau \in M$.

Example:



◆

Proof:

- Given an accepting run r construct a winning strategy f_0:

$$f_0(w, q) = (w, (r(w), t(w), r(w0), r(w1))$$

- Given a memoryless winning strategy f_0 construct an accepting run $r(\varepsilon) = s_0$ $\forall w \in \{0, 1\}^*$
  - $r(w0) = q$ where $f_0(w, r(w)) = (w, (\_, \_, q, \_))$
  - $r(w1) = q$ where $f_0(w, r(w)) = (w, (\_, \_, \_, q))$

■

Lemma 2 *For each parity tree automaton $\mathcal{A}$ over Σ-trees there exists a parity tree automaton $\mathcal{A}'$ over $\{1\}$-trees, such that $\mathcal{L}(\mathcal{A}) = \emptyset$ iff $\mathcal{L}(\mathcal{A}') = \emptyset$.*

Proof:

- $S' = S$;
- $s'_0 = s_0$;
- $M' = \{(q, 1, q_0.q_1) \mid (q, \sigma, q_0, q_1) \in M, \sigma \in \Sigma\}$
- $c' = c$

■

Theorem 3 *The language of a parity tree automaton $\mathcal{A} = (S, s_0, M, c)$ is non-empty iff Player 0 wins the parity game $\mathcal{G}_{\mathcal{A},t} = (V_0, V_1, E, c')$ from position s_0.*

- *$V_0 = S$;*
- *$V_1 = M$;*
- $E = \{(q, \tau) \mid \tau = (q, 1, q'_0, q'_1), \tau \in M\}$
  $\cup \{(\tau, q') \mid \tau = (q, 1, q'_0, q'_1)$ *and*
  $(q' = q'_0$ *or* $q' = q'_1)\}$;
- *$c'(q) = c(q)$ for $q \in S$;*
- *$c(\tau) = 0$ for $\tau \in M$.*