

Note del corso di Calcolabilità e Linguaggi Formali - Lezione 12

Alberto Carraro
14 dicembre 2016

IIS Bruno-Franchetti, Mestre (VE) <http://www.dsi.unive.it/~acarraro>

1 Parity games

Definition 1. A parity game is a tuple $\mathcal{G} = (V, R, \lambda, C, \Omega)$

- V is a finite set of states vertices partitioned into two sets V_V (vertices owned by "Verifier") and V_R (vertices owned by "Refuter"),
- $R \subseteq V \times V$ is the accessibility relation,
- C is a finite set of labels and $\lambda : V \times C$ is the labelling function,
- $\Omega : V \rightarrow \{1, \dots, k\}$ is the priority function.

A *play* is a finite or infinite sequence of vertices. A finite play π is winning for Verifier if its last vertex belongs to V_V . An infinite play $\pi = v_0, v_1, \dots$ is winning for Verifier if in the sequence $\Omega(\pi) = \Omega(v_0), \Omega(v_1), \dots$ the least priority that occurs infinitely often is even. We let

- $Win = \{\pi : \pi \text{ is a winning play for Verifier}\}$, so that $Win \subseteq V^* \cup V^\omega$.
- $Acc = \{\lambda(\pi) : \pi \in Win\}$, so that $Acc \subseteq C^* \cup C^\omega$.

Example 1. For a graph (V, R) and a finite set $F \subseteq C$. We can build a game such that $Acc \cap C^\omega = F^\omega$ by setting: $\Omega(v) = 0$ for every $v \in F$.

Example 2. For a graph (V, R) and a finite set $F \subseteq C$ we let $Reach_F$ be the set of plays π with $\pi = v_0, v_1, \dots \in V^\omega$ and such that there exists i with $\lambda(v_i) \in F$. We can build a game such that $Acc \cap C^\omega = \lambda(Reach_F)$ by setting:

2 Nondeterministic tree automata

Definition 2. A nondeterministic parity tree automaton (NPTA) for Σ -labelled full binary trees is a tuple $A = (Q, \Sigma, q_I, \delta, \Omega)$ where

- Q is a finite set of states,
- $q_I \in Q$ is the initial state,
- $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q^2)$ is the transition function and
- $\Omega : Q \rightarrow \{1, \dots, k\}$ is the priority function.

Remark 1. Note that the transition function $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q^2)$ is by some authors seen as a function $\delta : Q \times \Sigma \times \{0, 1\} \rightarrow \mathcal{P}(Q)$.

A *run-tree* of an automaton A over a tree t is a map $\rho : \{0, 1\}^* \rightarrow Q$ such that

1. $\rho(\varepsilon) = q_I$ and
2. for all $u \in \{0, 1\}^*$ we have $(\rho(u0), \rho(u1)) \in \delta(\rho(u), t(u))$.

A run-tree ρ of A over t is *accepting* iff in every path π of ρ the least priority that occurs infinitely often is even. An automaton A *accepts* a tree t just if there exists an accepting run-tree of A over t .

Definition 3. Given an NPTA $A = (Q, \Sigma, q_I, \delta, \Omega)$ and a tree t the acceptance parity game $\mathcal{G}_{A,t} = (V_V, V_R, E, v_I, \Omega')$ where

- $V_V = \{0, 1\}^* \times Q$
- $V_R = \{0, 1\}^* \times Q^2$
- $v_I = (\varepsilon, q_I)$
- $E \subseteq (V_V \times V_R) \cup (V_R \times V_V)$ is the smallest relation such that
 - if $(v, q) \in V_V$ and $(q_0, q_1) \in \delta(q, a)$ and $t(v) = a$, then $((v, q), (v, (q_0, q_1))) \in E$
 - if $(v, (q_0, q_1)) \in V_R$, then $((v, (q_0, q_1)), (v0, q_0)) \in E$ and $((v, (q_0, q_1)), (v1, q_1)) \in E$
- $\Omega'(v, q) = \Omega(q)$ and $\Omega'(v, (q_0, q_1)) = \max\{\Omega(q_0), \Omega(q_1)\}$

The acceptance parity game of Definition 3 can be read as follows:

- Verifier can exploit the nondeterminism of A by choosing the run-tree
- Refuter chooses the path

Their combined action is headed towards checking that there exists (\exists) a run-tree all (\forall) of whose paths are accepting. Again Verifier represents (\exists) and Refuter represents (\forall).

Lemma 1. Verifier has a winning strategy in $\mathcal{G}_{A,t}$ from vertex v_I iff $t \in L(A)$.

Proof. (\Rightarrow) Suppose Verifier has a winning strategy σ in $\mathcal{G}_{A,t}$ from v_I . Define the map $\rho : \{0, 1\}^* \rightarrow Q$ inductively as

- $\rho(\varepsilon) = q_I$;
- $\rho(v0) = q_0$ if $\rho(v) = q$ and $\sigma(v, q) = (v, (q_0, q_1))$;
- $\rho(v1) = q_1$ if $\rho(v) = q$ and $\sigma(v, q) = (v, (q_0, q_1))$.

Clearly ρ is a run-tree of A over t . A path π of ρ has vertices

$\dots, (v, q), (v, (q_0, q_1)), (v0, q_0), (v0, (q'_0, q'_1)), (v01, q'_1), \dots$

and therefore has priorities

$\dots, \Omega(q), \max\{\Omega(q_0), \Omega(q_1)\}, \Omega(q_0), \max\{\Omega(q'_0), \Omega(q'_1)\}, \Omega(q'_1), \dots$

The least priority occurring infinitely often is even but since each priority appearing at an even position is \geq than its immediate successor, the subsequence obtained by removing all priorities at even positions is such that the least priority occurring infinitely often is still even. Such a subsequence witnesses an accepting path of ρ and all paths of ρ are obtained this way. Hence ρ is accepting.

(\Leftarrow) Suppose ρ is an accepting run-tree of A over t . Then define the map $\sigma : (V_V \times V_R^*)V_V \rightarrow V_R$ as

- $\sigma(\alpha(v, q)) = (v, (q_0, q_1))$ iff $\rho(v0) = q_0$ and $\rho(v1) = q_1$.

Clearly σ is a strategy for Verifier in $\mathcal{G}_{A,t}$ from v_0 . A play π respecting σ has vertices

$\dots, (v, q), (v, (q_0, q_1)), (v0, q_0), (v0, (q'_0, q'_1)), (v01, q'_1), \dots$

and therefore has priorities

$\dots, \Omega(q), \max\{\Omega(q_0), \Omega(q_1)\}, \Omega(q_0), \max\{\Omega(q'_0), \Omega(q'_1)\}, \Omega(q'_1), \dots$

The subsequence obtained by removing all priorities at even positions is the run-tree ρ and the least priority occurring infinitely often in it is even. Since each priority appearing at an even position is \geq than its immediate successor, the least priority occurring infinitely often in π is still even. Such a play is winning for Verifier.

Definition 4. Given an NPTA $A = (Q, \Sigma, q_I, \delta, \Omega)$ the non-emptiness parity game $\mathcal{G}_A = (V_V, V_R, E, v_I, \Omega')$ where

- $V_V = Q$
- $V_R = \{((q, a), (q_0, q_1)) : (q_0, q_1) \in \delta(q, a)\}$.
- $v_I = q_I$
- $E \subseteq (V_V \times V_R) \cup (V_R \times V_V)$ is the smallest relation such that
 - if $q \in V_V$ and $(q_0, q_1) \in \delta(q, a)$, then $(q, ((q, a), (q_0, q_1))) \in E$
 - if $y = ((q, a), (q_0, q_1)) \in V_R$, then $(y, q_0) \in E$ and $(y, q_1) \in E$
- $\Omega'(q) = \Omega(q)$ and $\Omega'((q, a), (q_0, q_1)) = \Omega(q)$

Lemma 2. Verifier has a winning strategy in \mathcal{G}_A from vertex v_I iff $L(A) \neq \emptyset$.

Proof. (\Rightarrow) Suppose Verifier has a winning strategy σ in \mathcal{G}_A from v_I . Then $\delta(q, a) \neq \emptyset$ for every pair $(q, a) \in Q \times \Sigma$. The solution to the fixpoint equation $\tau(q) = (a, \tau(q_0), \tau(q_1))$ with constraints $\sigma(q) = ((q, a), (q_0, q_1))$ is a tree whose first coordinates define a tree $t \in L(A)$.

(\Leftarrow) Let $\alpha \in (V_V V_R)^* V_V$ be an initial segment of a play in \mathcal{G}_A from v_I . The run-trees ρ of the trees $t \in L(A)$. Suppose is an input tree and let ρ be an associated run-tree.

Let $\bar{t}(\varepsilon) = t(\varepsilon)$ and $\bar{t}(uj) = \bar{t}(u)t(uj)$ for $u \in \{0, 1\}^*$ and $j \in \{0, 1\}$. Let also $\bar{\rho}(\varepsilon) = \rho(\varepsilon)$ and $\bar{\rho}(uj) = \bar{\rho}(u)\rho(uj)$ for $u \in \{0, 1\}^*$ and $j \in \{0, 1\}$.

Come si fa a definire la strategia vincente? La run-tree su t potrebbe non visitare tutti gli stati e quindi per gli stati ne non compaiono come faccio ad avere la scelta giusta del successore per la strategia?

is a sequence $x_1(y_1, z_1), \dots, x_n(y_n, z_n)$ with $x_i \in Q$, $y_i \in Q \times \Sigma$ and $z_i \in Q^2$. The trace of α is then y_1, \dots, y_n . Then the map $\sigma : (V_V V_R)^* V_V \rightarrow V_R$ given by $\sigma(\alpha q) =$ component-wise first if α "contains" the string $u = i_1 \dots i_n$ and $t(i_1) \dots t(i_1 \dots i_n) = a$ and $\rho(i_1) \dots \rho(i_1 \dots i_n) = q$.