The Recursive Algorithm for Parity games

Material: "Recursive Solving of Parity Games Requires Exponential Time", Oliver Friedmann

Jeroen J.A. Keiren
jkeiren@win.tue.nl
http://www.win.tue.nl/∼jkeiren
HG 6.81

Department of Mathematics and Computer Science
Technische Universiteit Eindhoven

Parity games

Recall:

Definition (Parity game)

A parity game Γ is a four tuple $(V, E, p, (V_{Even}, V_{Odd}))$, where:

- (V, E) is a directed graph,
- E is a total edge relation,
- $p : V \to \mathbb{N}$ assigns priorities to vertices, and
- (V_{Even}, V_{Odd}) is a partitioning of V

- A player does a step in the game if a token is on a vertex owned by that player;
- A play (denoted π) is an infinite sequence of steps.

Notation

Let $G = (V, E, p, (V_{Even}, V_{Odd}))$ be a parity game.
We use the following notation:

- 0 for $Even$, 1 for Odd
- $1 - Even$ is Odd, $1 - Odd$ is $Even$
- $vE = \{v' \mid (v, v') \in E\}$
- $G \setminus U$ is parity game G restricted to the vertices outside U. Formally $G \setminus U = (V', E', p', (V'_{Even}, V'_{Odd}))$, with
  - $V' = V \setminus U$,
  - $E' = E \cap (V \setminus U)^2$,
  - $p'(v) = p(v)$ for $v \in V \setminus U$,
  - $V'_{Even} = V_{Even} \setminus U$, and
  - $V'_{Odd} = V_{Odd} \setminus U$

Strategies

- A strategy for $Player$ is a partial function $\psi_{Player}{:}V^* \times V_{Player} \rightarrow V$.
- A play $\pi = v_1 v_2 v_3 \ldots$ is consistent with strategy ψ_{Player} for $Player$ iff every $v_i \in \pi$ such that $v_i \in V_{Player}$ is immediately followed by $v_{i+1} = \psi_{Player}(v_1 \ldots v_i)$.

Definition (Memoryless strategy)

A memoryless strategy for $Player$ is a partial function $\psi_{Player}{:}V_{Player} \rightarrow V$ that decides the vertex the token is played to based on the current vertex.

Winning a parity game

Let $\pi = v_1 v_2 v_3 \ldots$ be a play:

- $\inf(\pi)$ denotes set of priorities occurring infinitely often in π;
- π is winning for player Even iff $\min(\inf(\pi))$ is even;

Definition (Winning strategy)

Strategy ψ_{Player} is a winning strategy for $Player$ from set $W \subseteq V$ if every play starting from a vertex in W, consistent with ψ_{Player} is winning for $Player$.

- There is a memoryless winning strategy for $Player$ from $W \subseteq V$ iff there is a winning strategy for $Player$ from W.

Goal

Let $G = (V, E, p, (V_{Even}, V_{Odd}))$ be a parity game.

- There is a unique partition (W_{Even}, W_{Odd}) of V such that:
  - $Even$ has winning strategy ψ_{Even} from W_{Even}, and
  - Odd has winning strategy ψ_{Odd} from W_{Odd}.

Goal of parity game algorithms

Compute partitioning (W_{Even}, W_{Odd}) with strategies ψ_{Even} and ψ_{Odd} of V, such that ψ_{Even} is winning for player $Even$ from W_{Even} and ψ_{Odd} is winning for player Odd from W_{Odd}.

Attractor sets

The attractor set for $Player$ and set $U \subseteq V$ is the set of vertices such that $Player$ can force any play to reach U.

Definition

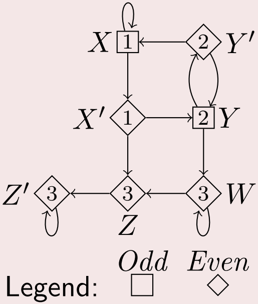
Let $U \subseteq V$. We define the attractor sets inductively as follows:

$$
\begin{aligned}
Attr^0_{Player}(G,U) &= U \\
Attr^{k+1}_{Player}(G,U) &= Attr^k_{Player}(G,U) \\
&\cup (V_{Player} \cap \{v \mid vE \cap Attr^k_{Player}(G,U) \neq \emptyset\}) \\
&\cup (V_{1-Player} \cap \{v \mid vE \subseteq Attr^k_{Player}(G,U)\}) \\
Attr_{Player}(G,U) &= \bigcup_{k \in \mathbb{N}} Attr^k_{Player}(G,U)
\end{aligned}
$$

Example of attractor sets

Example

Consider parity game G:



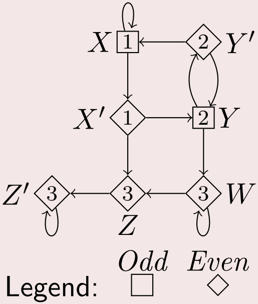
Compute:

- $Attr_0(G, \{Z\})$
- $Attr_1(G, \{W\})$

Example of attractor sets

Example

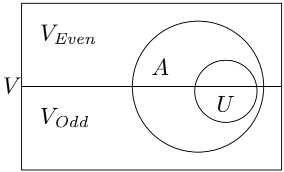
Consider parity game G:



Compute:

- $Attr_0(G, \{Z\})$
  $= \{Z, X', W\}$
- $Attr_1(G, \{W\}) = \{W, Y\}$

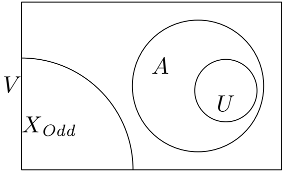
Observations

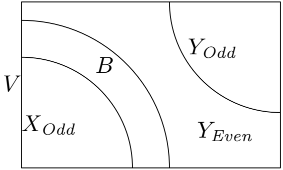


Let $U \subseteq V$. Let $A = Attr_{Even}(G, U)$.

- $Even$ cannot escape from $V \setminus A$. If it could, there would be an edge $(v, v') \in E$, such that $v \in V_{Even} \setminus A$, and $v' \in A$, but then by definition also $v \in A$, which is not the case.
- Odd cannot escape from A. If it could, there would be an edge $(v, v') \in E$, such that $v \in V_{Odd} \cap A$, and $v' \notin A$, but then by definition $v \notin A$.

Observations





Let $U \subseteq V$. Let $A = Attr_{Even}(G, U)$.
Assume:

- X_{Odd} is winning set for Odd on $G \setminus A$;
- $B = Attr_{Odd}(G, X_{Odd})$;
- Y_{Even} is winning set for $Even$ on $G \setminus B$;
- Y_{Odd} is winning set for Odd on $G \setminus B$.

Then:

- Player $Even$ can never leave B;
- Player Odd can never leave $V \setminus B$;
- A winning strategy for player Odd in $G \setminus (V \setminus B)$ from $V_{Odd} \cap B$ is also a winning strategy for player Odd in G from $V_{Odd} \cap B$.

Recursive algorithm (McNaughton '93, Zielonka '98)

Recursively solve a parity game: $Recursive(G)$. Returns partitioning (W_{Even}, W_{Odd}) such that $Even$ wins from W_{Even}, and Odd wins from W_{Odd}.

Base case:

if $V_G = \emptyset$ **then**
    $W_{Even} \leftarrow \emptyset$
    $W_{Odd} \leftarrow \emptyset$
    return (W_{Even}, W_{Odd})
end if

Inductive case (1):

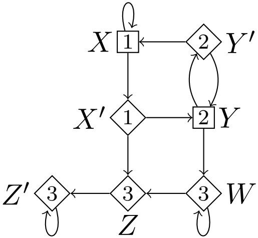
$m \leftarrow \min\{p(v) \mid v \in V\}$ (* Paper: max; assumes max parity game model, we use min parity games *)
$\mathit{Player} \leftarrow m \ \mathbf{mod}\ 2$
$U \leftarrow \{v \in V \mid p(v) = m\}$
$A \leftarrow \mathit{Attr}_{\mathit{Player}}(G, U)$
$(X_{\mathit{Even}}, X_{\mathit{Odd}}) \leftarrow \mathit{Recursive}(G \setminus A)$
if $X_{1-\mathit{Player}} = \emptyset$ **then**
  $W_{\mathit{Player}} \leftarrow A \cup X_{\mathit{Player}}$
  $W_{1-\mathit{Player}} \leftarrow \emptyset$
else
  ...
end if
return $(W_{\mathit{Even}}, W_{\mathit{Odd}})$

Inductive case (2):

$m \leftarrow \min\{p(v) \mid v \in V\}$
$\mathit{Player} \leftarrow m \textbf{ mod } 2$
$U \leftarrow \{v \in V \mid p(v) = m\}$
$A \leftarrow \mathit{Attr}_{\mathit{Player}}(G, U)$
$(X_{\mathit{Even}}, X_{\mathit{Odd}}) \leftarrow \mathit{Recursive}(G \setminus A)$
if $X_{1-\mathit{Player}} = \emptyset$ **then**
  $\ldots$
else
  $B \leftarrow \mathit{Attr}_{1-\mathit{Player}}(G, X_{1-\mathit{Player}})$
  $(Y_{\mathit{Even}}, Y_{\mathit{Odd}}) \leftarrow \mathit{Recursive}(G \setminus B)$
  $W_{\mathit{Player}} \leftarrow Y_{\mathit{Player}}$
  $W_{1-\mathit{Player}} \leftarrow B \cup Y_{1-\mathit{Player}}$
end if
return $(W_{\mathit{Even}}, W_{\mathit{Odd}})$

Example ($Recursive(G)$)

Consider parity game G:

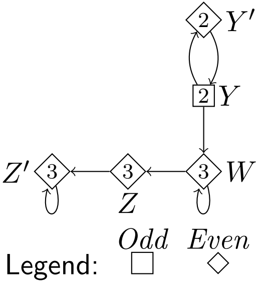


Legend: *Odd* □ *Even* ◇

$m \leftarrow 1$
$Player \leftarrow Odd$
$U \leftarrow \{v \in V \mid p(v) = 1\} = \{X, X'\}$
$A \leftarrow Attr_{Odd}(G, U) = \{X, X'\}$
$(X_{Even}, X_{Odd}) \leftarrow Recursive(G \setminus \{X, X'\})$

Example $(Recursive(G \setminus \{X, X'\}))$

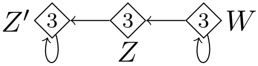
Consider parity game $G \setminus \{X, X'\}$:



6: $m \leftarrow 2$
7: $Player \leftarrow Even$
8: $U \leftarrow \{v \in V \setminus \{X, X'\} \mid p(v) = 2\} = \{Y, Y'\}$
9: $A \leftarrow Attr_{Even}(G \setminus \{X, X'\}, U) = \{Y, Y'\}$
10: $(X_{Even}, X_{Odd}) \leftarrow Recursive(G \setminus \{X, X', Y, Y'\})$

Example $(Recursive(G \setminus \{X, X', Y, Y'\}))$

Consider parity game $G \setminus \{X, X', Y, Y'\}$:

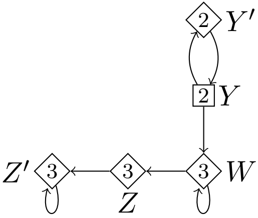


Legend: *Odd* □ *Even* ◇

$m \leftarrow 3$
$Player \leftarrow Odd$
$U \leftarrow \{v \in V \setminus \{X, X', Y, Y'\} \mid p(v) = 3\} = \{W, Z, Z'\}$
$A \leftarrow Attr_{Odd}(G \setminus \{X, X', Y, Y'\}, U) = \{W, Z, Z'\}$
$(X_{Even}, X_{Odd}) \leftarrow Recursive(G \setminus V) = (\emptyset, \emptyset)$
if $X_{Even} = \emptyset$ **then**
    $W_{Odd} \leftarrow A \cup X_{Odd} = A = \{W, Z, Z'\}$
    $W_{Even} \leftarrow \emptyset$
else
    ...
end if
return $(W_{Even}, W_{Odd}) = (\emptyset, \{W, Z, Z'\})$

Example ($Recursive(G \setminus \{X, X'\})$)

Consider parity game $G \setminus \{X, X'\}$:

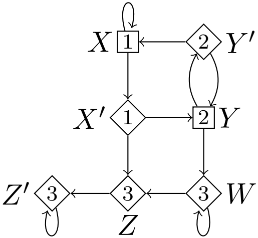


Legend: *Odd* □ *Even* ◇

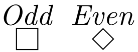
$m \leftarrow 2$
$Player \leftarrow Even$
$U \leftarrow \{v \in V \setminus \{X, X'\} \mid p(v) = 2\} = \{Y, Y'\}$
$A \leftarrow Attr_{Even}(G \setminus \{X, X'\}, U) = \{Y, Y'\}$
$(X_{Even}, X_{Odd}) \leftarrow Recursive(G \setminus \{X, X', Y, Y'\}) = (\emptyset, \{Z, Z', W\})$
if $X_{Odd} = \emptyset$ **then**
    ...
else
    $B \leftarrow Attr_{Odd}(G, X_{Odd}) = \{Y, Y', Z, Z', W\}$
    $(Y_{Even}, Y_{Odd}) \leftarrow Recursive(G \setminus V) = (\emptyset, \emptyset)$
    $W_{Even} \leftarrow Y_{Even} = \emptyset$
    $W_{Odd} \leftarrow B \cup Y_{Odd} = B = \{Y, Y', Z, Z', W\}$
end if
return $(W_{Even}, W_{Odd}) = (\emptyset, \{Y, Y', Z, Z', W\})$

Example ($Recursive(G)$)

Consider parity game G:



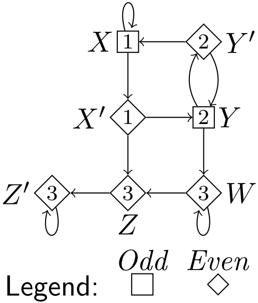
Legend:



$m \leftarrow 1$
$Player \leftarrow Odd$
$U \leftarrow \{v \in V \mid p(v) = 1\} = \{X, X'\}$
$A \leftarrow Attr_{Odd}(G, U) = \{X, X'\}$
$(X_{Even}, X_{Odd}) \leftarrow Recursive(G \setminus \{X, X'\}) = (\emptyset, \{Y, Y', Z, Z', W\})$
if $X_{Even} = \emptyset$ **then**
    $W_{Odd} \leftarrow A \cup X_{Even} = \{X, X', Y, Y', Z, Z', W\}$
    $W_{Even} \leftarrow \emptyset$
else
    ...
end if
return $(W_{Even}, W_{Odd}) = (\emptyset, \{X, X', Y, Y', Z, Z', W\})$

Example ($Recursive(G)$)

Consider parity game G:



So, player Odd wins from all vertices!

Complexity

Let $G = (V, E, p, (V_{Even}, V_{Odd}))$ be a parity game;
$n = |V|, e = |E|, d = \max\{p(v) \mid v \in V\}$.

Worst-case running time complexity:

$$\mathcal{O}(e \cdot n^d)$$

Lowerbound on worst-case:

$$\Omega(fib(n)) = \Omega((\frac{1+\sqrt{5}}{2})^n)$$

Complexity

Let $G = (V, E, p, (V_{Even}, V_{Odd})$ be a parity game;
$n = |V|, e = |E|, d = \max\{p(v) \mid v \in V\}$.

- Algorithm with best known upper bound: Big step algorithm due to Schewe, with complexity

$$\mathcal{O}(d \cdot n^{d/3})$$

- Big step combines recursive algorithm with small progress measures;
- Small progress measures will be discussed first lecture in January

Exercise

Consider the following parity game:

- Compute the winning sets W_{Even}, W_{Odd} for players $Even$ and Odd in this parity game using the recursive algorithm.
- Translate this parity game to BES and solve the BES using Gauss elimination.