1 Recursive algorithm

1.1 Original Zielonka's recursive algorithm

The original Zielonka's recursive algorithm,created from the constructive prove given in [1], is defined for a total PG (ie. an infinite game).

Algorithm 1 RecursivePG($PG\ G = (V, V_0, V_1, E, \Omega)$)

$m \leftarrow \min\{\Omega(v) \mid v \in V\}$
$h \leftarrow \max\{\Omega(v) \mid v \in V\}$
if $h = m$ or $V = \emptyset$ **then**
  if h is even or $V = \emptyset$ **then**
    return $(V, \emptyset)$
  else
    return $(\emptyset, V)$
  end if
end if
$\alpha \leftarrow 0$ if h is even and 1 otherwise
$U \leftarrow \{v \in V \mid \Omega(v) = h\}$
$A \leftarrow \alpha\text{-}Attr(G, U)$
$(W'_0, W'_1) \leftarrow$ RecursivePG$(G\backslash A)$
if $W'_{\overline{\alpha}} = \emptyset$ **then**
  $W_\alpha \leftarrow A \cup W'_\alpha$
  $W_{\overline{\alpha}} \leftarrow \emptyset$
else
  $B \leftarrow \overline{\alpha}\text{-}Attr(G, W'_{\overline{\alpha}})$
  $(W''_0, W''_1) \leftarrow$ RecursivePG$(G\backslash B)$
  $W_\alpha \leftarrow W''_\alpha$
  $W_{\overline{\alpha}} \leftarrow W''_{\overline{\alpha}} \cup B$
end if
return (W_0, W_1)

Definition 1.1. *[1] Given a parity game $G = (V, V_0, V_1, E, \Omega)$ and $U \subseteq V$ we define the subgame $G\backslash U$ to be the game $(V', V'_0, V'_1, E', \Omega)$ with:*

- $V' = V\backslash U$,
- $V'_0 = V_0 \cap V'$,
- $V'_1 = V_1 \cap V'$ *and*
- $E' = E \cap (V' \times V')$.

Definition 1.2. *[1] A set $U \subseteq V$ is an α-trap for PG G iff:*

$$\begin{aligned}
&\forall v \in U: \\
&\quad v \in V_\alpha \implies \forall (v, w) \in E : w \in U \\
&\quad \wedge \\
&\quad v \in V_{\overline{\alpha}} \implies \exists (v, w) \in E : w \in U
\end{aligned}$$

If the token is in α-trap U then player $\overline{\alpha}$ can play a strategy such that the token always remains in U.

Definition 1.3. *[1] For a non-empty set $U \subseteq V$ we define α-$Attr(G, U)$ such that*

$$U_0 = U$$

For $i \geq 0$:

$$\begin{aligned}
U_{i+1} = &\{v \in V_\alpha \mid \exists v' \in V : v' \in U_i \wedge (v, v') \in E\} \\
\cup &\{v \in V_{\overline{\alpha}} \mid \forall v' \in V : (v, v') \in E \implies v' \in U_i\}
\end{aligned}$$

$$\alpha\text{-}Attr(G, U) = U_k$$

such that for k we have

$$U_k = U_{k+1}$$

Lemma 1.1. *[1] Given PG $G = (V, V_0, V_1, E, \Omega)$, set $X \subseteq V$ and player α it holds that $V\backslash\alpha$-$Attr(G, X)$ is an α-trap in G.*

1.2 Recursive algorithm for VPGs

Definition 1.4. *A multidimensional parity game (MPG) is a parity game defined over a set of configuration $\mathfrak{C}$ and a set of origin vertices $\mathfrak{V}$. Every vertex is represented by a pair of a configuration and an origin vertex. We have MPG being a tuple $G = (V, V_0, V_1, E, \Omega)$ such that:*

- $V \subseteq \mathfrak{C} \times \mathfrak{V}$,
- $V_0 \uplus V_1 = V$,
- $E \subseteq V \times V$ *such that* $((c, v), (c', v')) \in E \implies c = c'$ *and*
- $\Omega : \mathfrak{V} \to \mathbb{N}$.

MPGs are total, ie. for every $(c, v) \in V$ there exists an $(c, w) \in V$ such that $((c, v), (c, w)) \in E$.

Definition 1.5. *For a set $X \subseteq \mathfrak{C} \times \mathfrak{V}$ we define $con(X) \subseteq \mathfrak{C}$ to be the set of configurations that occur in X, formally: $con(X) = \{c \mid (c, v) \in X\}$.*

Definition 1.6. *For a set $X \subseteq \mathfrak{C} \times \mathfrak{V}$ we define $cX = \{v \in \mathfrak{V} \mid (c, v) \in X\}$ and $Xv = \{c \in \mathfrak{C} \mid (c, v) \in X\}$.*

Definition 1.7. *For a set $E \subseteq (\mathfrak{C} \times \mathfrak{V}) \times (\mathfrak{C} \times \mathfrak{V})$ we define $cE = \{(v, w) \mid ((c, v), (c, w)) \in E\}$.*

Definition 1.8. *An MPG $G = (V, V_0, V_1, E, \Omega)$ can be played for a configuration $c \in \mathfrak{C}$ which is playing PG $(cV, cV_0, cV_1, cE, \Omega)$, we denote this as cG.*

Note that it follows immediately that for every $c \in \mathfrak{C}$ and MPG G defined over $\mathfrak{C}$ it holds that game cG is total, ie. for every $v \in cV$ there exists an $w \in cV$ such that $(v, w) \in E$.

Definition 1.9.

$$U_0 = U$$

For $i \geq 0$:

$$\begin{aligned} U_{i+1} = &\{(c, v) \in V_\alpha \mid \exists (c, v') \in V : (c, v') \in U_i \wedge ((c, v), (c, v')) \in E\} \\ &\cup \{(c, v) \in V_{\overline{\alpha}} \mid \forall (c, v') \in V : ((c, v), (c, v')) \in E \implies (c, v') \in U_i\} \end{aligned}$$

$$\alpha\text{-}MAttr(G, U) = U_k$$

such that for k we have

$$U_k = U_{k+1}$$

Definition 1.10. $\setminus^M : MPG \to \mathcal{P}(V) \to MPG$
$(V, V_0, V_1, E, \Omega) \setminus^M CV = (V', V_0', V_1', E', \Omega)$ *such that:*
$V' = V \setminus CV$
$E' = E \cap (V' \times V')$
$V_0' = V_0 \cap V'$
$V_1' = V_1 \cap V'$

Definition 1.11. *A set $U \subseteq V$ is a α-MTrap for MPG G iff:*

$$\begin{aligned} &\forall (c, v) \in U : \\ &\qquad (c, v) \in V_\alpha \implies \forall ((c, v), (c, w)) \in E : (c, w) \in U \\ &\qquad \wedge \\ &\qquad (c, v) \in V_{\overline{\alpha}} \implies \exists ((c, v), (c, w)) \in E : (c, w) \in U \end{aligned}$$

For the following lemma's consider MPG $G = (V, V_0, V_1, E, \Omega)$ defined over $\mathfrak{C}$ and $\mathfrak{V}$. We write U and X to denote subsets of V.

Lemma 1.2. *If $(c, v) \in \alpha{-}MAttr(G, U)$ then there exists a $(c, x) \in U$ for any $c \in \mathfrak{C}$.*

Proof. Let $(c, v) \in \alpha\text{-}MAttr(G, U)$. If $(c, v) \in U$ then the lemma holds. Consider $(c, v) \notin U$. Using definition 1.9 we can conclude that $(c, v) \in U_k$.

We will prove that for $i \geq 0$ it holds that if $(c, x) \in U_{i+1}$ for some x then $(c, x') \in U_i$ for some x'.

Let $(c, x) \in U_{i+1}$ with $i \geq 0$. If $(c, x) \in V_\alpha$ we know that $((c, x), (c, x')) \in E$ and $(c, x') \in U_i$ by definition 1.9. If $(c, x) \in V_{\overline{\alpha}}$ we know that for all successors (c, x') of (c, x) we have $(c, x') \in U_i$. Since there exists at least 1 successor (MPGs are total, using definition 1.9) we know that there exists a $(c, x') \in U_i$.

Since $(c, v) \in U_k$ we know there exists a $(c, w) \in U_l$ for every $l < k$, this includes $l = 0$ so we can conclude that for some w we have $(c, w) \in U_0 = U$, proving the lemma. □

Algorithm 2 RecursiveMPG(*MPG* $G = (V, V_0, V_1, E, \Omega)$)

$m \leftarrow \min\{\Omega(v) \mid (c, v) \in V\}$
$h \leftarrow \max\{\Omega(v) \mid (c, v) \in V\}$
if $h = m$ or $V = \emptyset$ **then**
    if h is even or $V = \emptyset$ **then**
        return $(V, \emptyset)$
    else
        return $(\emptyset, V)$
    end if
end if
$\alpha \leftarrow 0$ if h is even and 1 otherwise
$U \leftarrow \{(c, v) \in V \mid \Omega(v) = h\}$
$A \leftarrow \alpha\text{-}MAttr(G, U)$
$(W'_0, W'_1) \leftarrow$ RecursiveMPG($G \setminus^M A$)
if $W'_{\overline{\alpha}} = \emptyset$ **then**
    $W_\alpha \leftarrow A \cup W'_\alpha$
    $W_{\overline{\alpha}} \leftarrow \emptyset$
else
    $B \leftarrow \overline{\alpha}\text{-}MAttr(G, W'_{\overline{\alpha}})$
    $(W''_0, W''_1) \leftarrow$ RecursiveMPG($G \setminus^M B$)
    $W_\alpha \leftarrow W''_\alpha$
    $W_{\overline{\alpha}} \leftarrow W''_{\overline{\alpha}} \cup B$
end if
return (W_0, W_1)

Lemma 1.3. *The set $U \subseteq V$ is an α-MTrap in the game G iff for all $c \in con(V) = \mathfrak{C}$ the set cU is an α-trap in the game cG.*

Proof. We first note the following equivalences:

- $(c, v) \in X$ iff $v \in cX$ for some $X \subseteq V$ (using 1.6).
- $((c, v), (c, w)) \in E$ iff $(v, w) \in cE$ (using 1.7).

We can now rewrite the MTrap definition (1.11):

$$\begin{aligned} &\forall c \in con(U) : \forall v \in cU : \\ &\qquad v \in cV_\alpha \implies \forall (v, w) \in cE : w \in cU \\ &\qquad \wedge \\ &\qquad v \in cV_{\overline{\alpha}} \implies \exists (v, w) \in cE : w \in cU \end{aligned}$$

Using the trap definition (1.2) on a game cG we can rewrite the MTrap definition to:

$$\forall c \in con(U) : cU \text{ is an } \alpha\text{-trap in } cG$$

To prove the lemma we first consider the set $U \subseteq V$ that is an α-MTrap in game G and some $c \in con(V)$. If $c \in con(U)$ then using our rewritten definition we conclude that cU is an α-trap in cG. If $c \notin con(U)$ then $cU = \emptyset$, the empty set is an α-trap by definition. So in either case $c \in con(V)$ is an α-trap in cG.

Next we consider $U \subseteq V$ where for every $c \in con(V)$ the set cU is an α-trap in the game cG. Because $U \subseteq V$ we have for every $c \in con(U)$ the set cU is an α-trap in the game cG. Using our rewritten MTrap definition we can conclude that U is an α-MTrap in G. □

Lemma 1.4. *$X = \alpha\text{-}MAttr(G, U)$ iff $cX = \alpha\text{-}Attr(cG, cU)$ for all $c \in con(U)$.*

Proof. Consider the MAttr definition (1.9). To calculate U_{i+1} all vertices $(c, v) \in V$ are considered, however (c, v) can only be attracted if there is an edge from (c, v) to $(c, v') \in U_i$. Because two vertices that are connected by an edge have the same configuration we only to consider vertices $(c, v) \in V$ for $c \in U_i$. We can also conclude that $con(U) = con(U_i)$ for any $i \geq 0$.

We can now rewrite the MAttr definition to the following:

For all $c \in con(U)$ we have:

$$cU_0 = cU$$

For $i \geq 0$:

$$cU_{i+1} = \{v \in cV_\alpha \mid \exists v' \in cV : v' \in cU_i \wedge (v, v') \in cE\}$$
$$\cup \{v \in cV_{\overline{\alpha}} \mid \forall v' \in cV : (v, v') \in cE \implies v' \in cU_i\}$$

$$c(\alpha\text{-}MAttr(G, U)) = cU_k$$

such that for k we have

$$cU_k = cU_{k+1}$$

This definition is identical to the α-Attr definition filled in for game cG and set cU, therefore the lemma holds. □

Lemma 1.5. *The set $U = V \backslash \alpha{-}MAttr(G, X)$ is an α-MTrap in G for any $X \subseteq V$.*

Proof. Assume U is not an α-MTrap. Using the MTrap definition (1.11) we can conclude that there exists an $(c, v) \in U$ such that either:

1. $(c, v) \in V_\alpha \wedge \exists((c, v), (c, w)) \in E : (c, w) \notin U$ or
2. $(c, v) \in V_{\overline{\alpha}} \wedge \forall((c, v), (c, w)) \in E : (c, w) \notin U$

Note that $(c, w) \notin U$ is equivalent to $(c, w) \in \alpha\text{-}MAttr(G, X)$.

If $(c, v) \in V_\alpha$ then there exists an $((c, v), (c, w)) \in E$ such that $(c, w) \in \alpha\text{-}MAttr(G, X)$. Using the MTrap definition (1.11) the vertex (c, v) will also be attracted and therefore $(c, v) \in \alpha\text{-}MAttr(G, X)$ and $(c, v) \notin U$ which is a contradiction.

If $(c, v) \in V_{\overline{\alpha}}$ then for all $((c, v), (c, w)) \in E$ we have $(c, w) \in \alpha\text{-}MAttr(G, X)$. Using the MTrap definition (1.11) the vertex (c, v) will also be attracted and therefore $(c, v) \in \alpha\text{-}MAttr(G, X)$ and $(c, v) \notin U$ which is a contradiction.

In all cases we derive at a contradiction, therefore U is an α-MTrap. □

Lemma 1.6. *If $V \backslash U$ is an α-MTrap then $G \backslash U$ is an MPG.*

Proof. To show that $G \backslash U$ is an MPG we have to prove its completeness. Let $G \backslash U = G' = (V', V'_0, V'_1, E', \Omega)$ (using 1.10). Assume G' is not complete, there exists an $c \in \mathfrak{C}$ such that game cG' is not complete. There exists an $v \in cV'$ such that v has no outgoing edges. Since cG is complete vertex v does have an outgoing edge in cG. So v h as an outgoing edge that goes to cU, furthermore all the outgoing edges from v go to cU. Since $v \in cV \backslash cU$ we can conclude that we are forced to leave $cV \backslash cU$ when the token is on vertex v, therefore $cV \backslash cU$ is not an α-trap and $V \backslash U$ is not an α-MTrap. This is a contradiction therefore our assumption is wrong and G' is complete. □

Lemma 1.7. *Let $X \subseteq V$ be an α-MTrap in G. Then $U = \overline{\alpha}{-}MAttr(G, X)$ is also an α-MTrap in G.*

Proof. Using lemma 1.3 we find that for all $c \in \mathfrak{C}$ the set cX is an α-trap in cG. Furthermore, using lemma 1.4, $cU = \overline{\alpha}\text{-}Attr(cG, cX)$ for all $c \in con(X)$.

Consider $c \in \mathfrak{C}$. If $c \in con(X)$ then using Zielonka's lemma 1.1 we find that cU is an α-trap in G. If $c \notin con(X)$ then $cU = \emptyset$ which is an α-trap in G. So for any $c \in \mathfrak{C}$ the set cU is an α-trap. Therefore we can apply lemma 1.3 to find that U is an α-MTrap. □

Lemma 1.8. *Let $X \subseteq V$. If $V \backslash X$ is an α-MTrap in G and $X' \subseteq V \backslash X$ is an α-MTrap in $G \backslash^M X$ then X' is an α-MTrap in G.*

Proof. Let $G' = (V', V'_0, V'_1, E', \Omega) = G \backslash^M X$. Consider vertex $(c, v) \in X'$, we distinguish two cases:

- If $(c, v) \in V'_\alpha$, there exists an outgoing edge from (c, v) inside X' for game G', this edge still exists for game G.
- If $(c, v) \in V_{\overline{\alpha}}$, all outgoing edges from (c, v) are inside X' for game G'. Since $(c, v) \in V \backslash X$ we know that all outgoing edges are inside $V \backslash X$ in game G. So (c, v) has the same outgoing edges in game G as in game G', therefore all its outgoing edges are inside X' for game G.

This proves that X' is an α-MTrap in G. □

Theorem 1.9. *Given MPG $G = (V, V_0, V_1, E, \Omega)$ it holds that $(c, v) \in W_\alpha$ resulting* RecursiveMPG(G) *iff player α has a winning memoryless strategy in cG.*

Proof. Proof by induction, similar to [1].

Induction hypothesis (IH):

For $(W_0, W_1) = $ RecursiveMPG$(G = (V, V_0, V_1, E, \Omega))$ we have

1. $W_0 \uplus W_1 = V$,
2. for any $\alpha \in \{0, 1\}$ it holds that W_α is an $\overline{\alpha}$-MTrap in G and

3. for every $c \in con(W_\alpha)$ there is a strategy σ_α^c such that $v \in cW_\alpha$ is winning for player α in game cG.

We will refer to the parts of the IH as IH1, IH2 and IH3.

Base $\max\{\Omega(v) \mid (c,v) \in V\} = \min\{\Omega(v) \mid (c,v) \in V\}$:

There is only one priority, so any infinite play for any configuration is won by the player with the parity of this one priority. So the entire graph is won by one player (proving IH1), it is a α-MTrap for any $\alpha \in \{0,1\}$ and the winner of the graph is not affected by the strategies (proving IH2 and IH3). In line 1-9 of the algorithm this is implemented, so in this case the IH holds.

Base $V = \emptyset$:

An empty set is trivially an α-MTrap so returning $(\emptyset, \emptyset)$ satisfies the IH. This is implemented in line 1-5 in the algorithm.

Step:

Let α be 0 if the highest priority in the graph in even and 1 otherwise. (line 10)

Using line 11 and 12 and lemma 1.5 we get that $V \backslash A$ is an α-MTrap in G.

Using lemma 1.6 we find that $G \backslash^M A$ is an MPG. Since U is non-empty A is non-empty and therefore $G \backslash^M A$ is smaller than G. Therefore we can apply the IH on it and we find W_0' and W_1'. Let the associated strategies be w_0^c and w_1^c.

We distinguish two cases:

- $W_{\overline{\alpha}} = \emptyset$:

  We claim that sets $W_\alpha = W_\alpha' \cup A$ and $W_{\overline{\alpha}} = \emptyset$ satisfy the IH. Clearly $W_\alpha = V$, so the winning sets are the entire graph and the empty set which are both an α-MTrap and an $\overline{\alpha}$-MTrap trivially (proving IH1 and IH2).

  To prove IH3 we will consider game cG for any $c \in con(W_\alpha) = con(V)$. By showing that player α has a winning strategy from every cV we prove IH3.

  Consider play π in game cG and strategy σ_α^c that plays towards cU when the token is in $cA \backslash cU$, plays w_α^c when the token is in $cV \backslash cA$ and plays an arbitrary edge when the token is in cU.

  Since cA is an attractor the token will always reach cU when in cA and σ_α^c is played. So the token can only escape cA through cU. Consider the token being in $cV \backslash cA$, if player $\overline{\alpha}$ plays to stay in $cV \backslash cA$ then player α wins, since strategy w_α^c is winning for every vertex in $cV \backslash cA$ if the token doesn't escape. If the token is played towards cA by player $\overline{\alpha}$ then the play can eventually return to $cV \backslash cA$ in which case cU is visited or the play can remain inside cA in which case cU is visited infinitely often. So a play can stay in $cV \backslash cA$ in which case player α wins, can play towards and stay in cA in which case player α wins or alternate between the two in which case cU is infinitely often visited and player α wins.

  This is implemented in line 14-16 of the algorithm.

- $W_{\overline{\alpha}} \neq \emptyset$:

  Recall that $V \backslash A$ is an α-MTrap in G, by IH we know that $W_{\overline{\alpha}}'$ is an α-MTrap in $G \backslash^M A$, therefore (using lemma 1.8) $W_{\overline{\alpha}}'$ is an α-MTrap in G.

  Let $B = \overline{\alpha} - MAttr(G, W_{\overline{\alpha}}')$ (line 18). , using lemma 1.7 we know that B is also an α-MTrap.

  By lemma 1.6 we find that $G \backslash^M B$ is an MPG. Since B is non-empty the game $G \backslash^M B$ is smaller than the game G, therefore we can apply the IH and we find W_0'' and W_1''. Let the associated strategies be q_0^c and q_1^c. Finally let $W_\alpha = W_\alpha''$ and $W_{\overline{\alpha}} = W_{\overline{\alpha}}'' \cup B$ (lines 18-21).

  Since $W_\alpha'' \uplus W_{\overline{\alpha}}'' = V \backslash B$ by IH we have $W_\alpha \uplus W_{\overline{\alpha}} = V$ (proving IH1).

  $V \backslash B$ is an $\overline{\alpha}$-MTrap in G by lemma 1.5.

  W_α'' is an $\overline{\alpha}$-MTrap in $G \backslash^M B$ by IH.

  $W_\alpha = W_\alpha''$ is an $\overline{\alpha}$-MTrap in G because it is an $\overline{\alpha}$-MTrap in $G \backslash^M B$ and $V \backslash B$ is an $\overline{\alpha}$-MTrap in G (using lemma 1.8). (proving IH2 for α)

  $W_{\overline{\alpha}}''$ is an α-MTrap in $G \backslash^M B$ by IH.

  So $\overline{\alpha}$ has a strategy such that the token can not go from $W_{\overline{\alpha}}''$ to W_α'' directly.

  Player $\overline{\alpha}$ can make sure $W_{\overline{\alpha}}''$ can only be left by going to B. In B player $\overline{\alpha}$ has a strategy to stay in B so player $\overline{\alpha}$ can force the token to stay inside $W_{\overline{\alpha}} = W_{\overline{\alpha}}'' \cup B$, hence it is an α-MTrap (proving IH2 for $\overline{\alpha}$).

  What is left to show is that for game cG for any $c \in con(W_\beta)$ player β has a winning strategy for cW_β with $\beta \in \{0,1\}$.

  Consider game cG, let the token be on vertex $v \in cV$. We distinguish three cases:

  - If $v \in cW_{\overline{\alpha}}'$. We know that $cW_{\overline{\alpha}}'$ is an α-trap in cG because $V \backslash A$ is also an α-trap and player $\overline{\alpha}$ has a winning strategy for every v.
  - If $v \in cB \backslash cW_{\overline{\alpha}}'$. Player $\overline{\alpha}$ has a strategy such that the token eventually ends up in $cW_{\overline{\alpha}}'$ and therefore player $\overline{\alpha}$ wins.

– If $v \in cW''_{\overline{\alpha}}$. As shown above player $\overline{\alpha}$ has a strategy such that the token remains in $cW''_{\overline{\alpha}}$ or goes to cB. In the latter case player $\overline{\alpha}$ wins as shown above. In the first case player $\overline{\alpha}$ also wins by playing strategy $q^c_{\overline{\alpha}}$.

– if $v \in cW''_{\alpha}$. Since cW''_{α} is an $\overline{\alpha}$-trap in cG, player α can play strategy q^c_{α} such that the token remains in cW''_{α} and player α wins.

This proves IH3.

□

Theorem 1.10. *Given VPG $G = (V, V_0, V_1, E, \Omega, \mathfrak{C}, \theta)$ with winning sets Q^c_0 and Q^c_1. It holds that $v \in Q^c_\alpha$ iff $(c, v) \in W_\alpha$ resulting from* RecursiveMPG$(V', V'_0, V'_1, E', \Omega)$ *where:*

- $V' = \mathfrak{C} \times V$,
- $V'_0 = \mathfrak{C} \times V_0$,
- $V'_1 = \mathfrak{C} \times V_1$,
- $E' = \{((c, v), (c, w)) \mid (v, v') \in E \wedge c \in \theta(v, w)\}$

References

[1] W. Zielonka, "Infinite games on finitely coloured graphs with applications to automata on infinite trees," *Theoretical Computer Science*, vol. 200, no. 1, pp. 135 – 183, 1998.