Chapter 4

Infinite Two-Player Games with Perfect Information: Applications

4.1 Büchi and Muller Acceptance Conditions

4.1.1 Deterministic Automata

Remark 36 (1) Every language $L(\mathcal{A})$ recognized by a Deterministic Büchi Automaton $\mathcal{A} = (\Sigma, Q, q_i, \delta, \mathcal{F})$ is $\mathbf{\Pi}^\circ_2$.

(2) There exists some Deterministic Büchi Automaton $\mathcal{A} = (\Sigma, Q, q_i, \delta, \mathcal{F})$ such that $L(\mathcal{A})$ is $\mathbf{\Pi}^\circ_2$-complete.

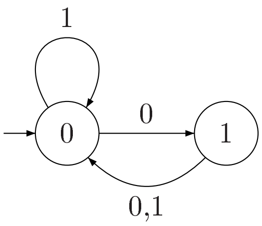
Proof of remark 36:

(1) Given a finite word $u \in \Sigma^*$ a state $q \in Q$, we write $\mathcal{A}(q, u) = q'$ to denote the state q' reached from q after reading u.

$$\begin{aligned}
x \in L(\mathcal{A}) \quad & \iff \quad \exists q \in \mathcal{F} \;\; \forall n \;\; \exists m > n \;\; \mathcal{A}(q_i, x_0 \dots x_m) = q \\
& \iff \quad \forall n \;\; \exists m > n \;\; \exists q \in \mathcal{F} \;\; \mathcal{A}(q_i, x_0 \dots x_m) = q \\
& \iff \quad x \in \underbrace{\bigcap_{n \in \mathbb{N}} \underbrace{\bigcup_{m>n,\; q \in \mathcal{F}} \underbrace{W_{q,m} \cdot \Sigma^\omega}_{\mathbf{\Sigma}^\circ_1}}_{\mathbf{\Sigma}^\circ_1}}_{\mathbf{\Pi}^\circ_2}
\end{aligned}$$

where $W_{q,m}$ is the set of all finite words u on Σ of length precisely m such that the automaton $\mathcal{A}$ starting in the initial state q_i stops at state q after reading u.

(2) Consider the following Deterministic Büchi Automaton $\mathcal{A}$:



$$\mathcal{F} = \{1\}$$

It recognizes the words with infinitely many 0. One can easily show that $I\!I$ has a w.s. in the game $\mathbf{W}\left((\oplus \xrightarrow{0} \ominus)^{\sim 1}\ ,\ \mathcal{A}\right)$.

$\dashv$ 36

Definition 37 *A Deterministic Muller Automaton is an automaton $\mathcal{A} = (\Sigma, Q, q_i, \delta, \mathcal{F})$ such that:*

(1) Σ is the alphabet ($\Sigma \neq \emptyset$),

(2) Q the set of states ($Q \neq \emptyset$),

(3) q_i is the initial state,

(4) $\delta :\ \Sigma \times Q \longmapsto Q$ is the transition function,

(5) $\mathcal{F} \subseteq \mathcal{P}(Q)$ is the set of acceptance conditions.

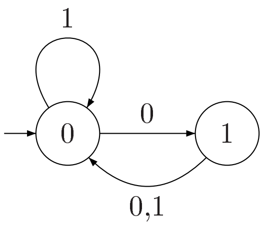
Given a word $x \in \Sigma^\omega$, This word x is accepted by $\mathcal{A}$ precisely when $Inf_{\mathcal{A}}(x) \in \mathcal{F}$.

Examples 38

(1) Every language $L(\mathcal{B})$ that is accepted by a Deterministic Büchi Automaton $\mathcal{B} = (\Sigma, Q, q_i, \delta, \mathcal{F}_B)$ is recognized by the Deterministic Muller Automaton $\mathcal{A} = (\Sigma, Q, q_i, \delta, \mathcal{F}_A)$ with $\mathcal{F}_A = \{Q' \subseteq Q :\ Q' \cap \mathcal{F}_B \neq \emptyset\}$.

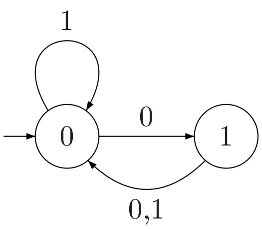
(2)

(a) $\mathcal{A} =$



$$\mathcal{F} = \{\{0\}\}$$

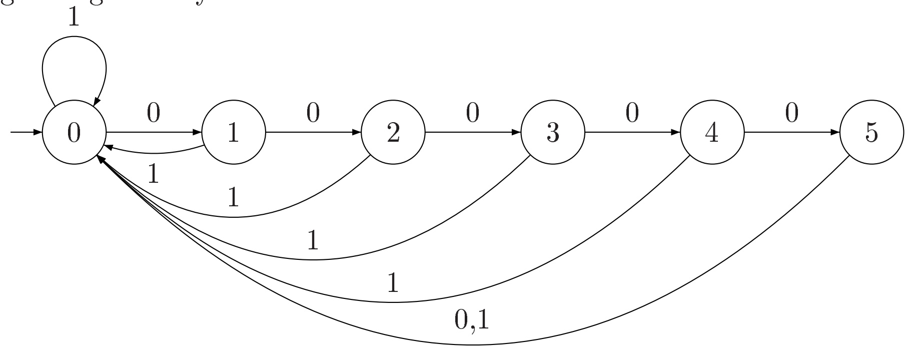
(b) Clearly, the language recognized by $\mathcal{B} =$



$$\mathcal{F} = \{\{0,1\}\}$$

is the complement of the language accepted by $\mathcal{A}$. And I has a w.s. in the game $\mathbf{W}\left(L(\mathcal{A})\ ,\ L(\mathcal{B})\right)$, namely, play 1 as first move and then copy $I\!I$'s run. Moreover, $L(\mathcal{B})$ is recognized by a Deterministic Büchi Automaton, and we already know that it is $\mathbf{\Pi}^\circ_2$-complete. The previous remark shows that $L(\mathcal{A})$ is not in $\mathbf{\Pi}^\circ_2$ and, since $L(\mathcal{B}) = \Sigma^\omega \smallsetminus L(\mathcal{A})$, $L(\mathcal{A}) \in \mathbf{\Sigma}^\circ_2$. In fact $L(\mathcal{A})$ is $\mathbf{\Sigma}^\circ_2$-complete.

(c) The language recognized by $\mathcal{C} =$



$$\mathcal{F} = \{\{0\}, \{0,1,2\}, \{0,1,2,3,4\}\}$$

is not in $\mathbf{\Sigma}^\circ_2$ (I has a w.s. in $\mathbf{W}\left(\mathcal{C}\ ,\ \mathcal{A}\right)$) nor in $\mathbf{\Pi}^\circ_2$ (I also has a w.s. in $\mathbf{W}\left(\mathcal{C}\ ,\ \mathcal{B}\right)$). But it is in both $\mathbf{\Sigma}^\circ_3$ and $\mathbf{\Pi}^\circ_3$:

For each state $q \in Q$, let $\mathcal{C}_q$ be the Deterministic Büchi Automaton defined exactly as $\mathcal{C}$, except that the acceptance conditions are x is accepted iff $q \in Inf_{\mathcal{C}}(x)$. Clearly:

$$L(\mathcal{C}) = \left(\begin{array}{c} L(\mathcal{C}_0) \bigcap \Big(L(\mathcal{C}_1) \cup L(\mathcal{C}_2) \cup L(\mathcal{C}_3) \cup L(\mathcal{C}_4) \cup L(\mathcal{C}_5) \Big)^{\complement} \\ \bigcup \\ \Big(\big(L(\mathcal{C}_0) \cap L(\mathcal{C}_1) \cap L(\mathcal{C}_2) \big) \bigcap \big(L(\mathcal{C}_3) \cup L(\mathcal{C}_4) \cup L(\mathcal{C}_5) \big)^{\complement} \Big) \\ \bigcup \\ \Big(\big(L(\mathcal{C}_0) \cap L(\mathcal{C}_1) \cap L(\mathcal{C}_2) \cap L(\mathcal{C}_3) \cap L(\mathcal{C}_4) \big) \bigcap L(\mathcal{C}_5)^{\complement} \Big) \end{array} \right)$$

Since each $L(\mathcal{C}_i)$ belongs to $\mathbf{\Pi}^\circ_2$, the language recognized by $\mathcal{C}$ belongs to the class of the Boolean Combination of $\mathbf{\Pi}^\circ_2$ sets (i.e. the smallest class that contains $\mathbf{\Pi}^\circ_2$ sets and is closed under *finite* union and complementation. Since $\mathbf{\Pi}^\circ_2 \subseteq \mathbf{\Sigma}^\circ_3 \cap \mathbf{\Pi}^\circ_3$, $\mathbf{\Sigma}^\circ_2 \subseteq \mathbf{\Sigma}^\circ_3 \cap \mathbf{\Pi}^\circ_3$, and both $\mathbf{\Sigma}^\circ_3$ and $\mathbf{\Pi}^\circ_3$ are closed under *finite union* and *finite intersection*. Hence any set that is written as a Boolean Combination of $\mathbf{\Pi}^\circ_2$ sets stands in both $\mathbf{\Sigma}^\circ_3$ and $\mathbf{\Pi}^\circ_3$.

Of course, the result generalizes from $\mathcal{C}$ to any Deterministic Muller Automaton. Therefore, we have just showed that for any Deterministic Muller Automaton, the language it recognizes is a Boolean Combination of $\mathbf{\Pi}^\circ_2$ sets.

4.1.2 Deterministic Turing Machines

Instead of taking Automata into account, one can work with more refined Deterministic Machines like Pushdown Automata, or even Turing Machines. And for any kind of such machines, one can consider Büchi acceptance conditions or Muller acceptance conditions. In terms of topological complexity, the result remain the same. Namely:

Remark 39

(1) Every set recognized by a Deterministic Turing Machine with Büchi acceptance conditions belongs to $\mathbf{\Pi}^\circ_2$.

(2) Every set recognized by a Deterministic Turing Machine with Muller acceptance conditions belongs to the Boolean Combination of $\mathbf{\Pi}^\circ_2$ sets, hence is in $\mathbf{\Sigma}^\circ_3 \cap \mathbf{\Pi}^\circ_3$.

4.1.3 The Problem of Non Determinism

What happens when we still consider Büchi or Müller acceptance conditions, but with nondeterministic machines? Is the language recognized still Borel? The answer is: it depends on the kind of machine involved.

- Given any nondeterministic Büchi Automaton, one can build a Deterministic Muller Automaton that recognizes the very same language. Therefore, any nondeterministic Muller Automaton is also equivalent to some other Deterministic Muller Automaton. Hence the languages accepted are all Boolean Combinations of $\mathbf{\Pi}^\circ_2$ sets.

- This result does not hold as soon as we consider machines slightly more complicated like PushDown Automata. In fact, there exist nondeterministic PushDown Automata with Büchi acceptance Conditions such that the sets they recognize are not Borel. Moreover, one can even find a NPDA with the property that the language it recognizes is either a set that is not Borel or it is the set of all infinite words, but one cannot decide which case. As a consequence, there exists NPDA $\mathcal{A}$ such that one cannot decide who wins the game $\mathbb{G}(L(\mathcal{A}))$.

So, for the sequel, we forget about nondeterministic Machines. And for simplicity, we concentrate on infinite games played over finite graphs.

4.2 Strategies and Memory

In this section we consider games played on graphs with various winning conditions. All these games will be clearly determined. We ask the question whether there exists a winning strategy that does not need any memory, or if there is no such, what is the minimum memory that a winning strategy needs.

Definition 40 *An arena is a Game Graph $G = (V_0, V_1, E, C, \phi)$, where*

(1) V_0 and V_1 are non empty sets of vertices,

(2) $E \subseteq V_0 \times V_1 \cup V_1 \times V_0$ is the set of edges such that for each vertice $v \in V_0 \cup V_1$, the set $vE = \{v' \in V_0 \cup V_1 :\ (v, v') \in E\} \neq \emptyset$.

(3) C is a finite non empty set of colours.

(4) ϕ is a partial mapping: $V_0 \cup V_1 \longmapsto C$.

Two players (0 and 1) move a token alternatively between vertices (if the token is in a vertex $v \in V_0$ (resp. $\in V_1$). player 0 (resp. player 1) chooses $v' \in vE$. An infinite run induces an infinite path $v_o v_1 v_2 \ldots$,

(5) for each such play $v_o v_1 v_2 \ldots$, there are infinitely many i such that $\phi(v_i)$ is defined (v_i is coloured).

(6) (generalized) Muller conditions are given by two complementary collections of non empty subsets of C:

$$\mathcal{F}_0 \subseteq (\mathcal{P}(C) \setminus \{\emptyset\}), \quad \mathcal{F}_1 = (\mathcal{P}(C) \setminus \{\emptyset\}) \setminus \mathcal{F}_0$$

Player 0 wins the play $x = v_o v_1 v_2 \ldots$ if $\phi(Inf(x)) \in \mathcal{F}_0$, 1 wins otherwise.

If each node has a different color, then one obtains the usual definition of Muller acceptance conditions.
We already know that such games are determined, so that one of the two players has a w.s., which formally is a function σ that assigns a value $\sigma(v_o v_1 v_2 \ldots v_i) = v_{i+1}$ on all sequence $v_o v_1 v_2 \ldots v_i$ where i is a node for the right player.

Definition 41

(1) A finite memory *strategy for player i is a mapping $\sigma : V_i \times M \longmapsto V_{1-i}$ where M is a finite set called the* memory.

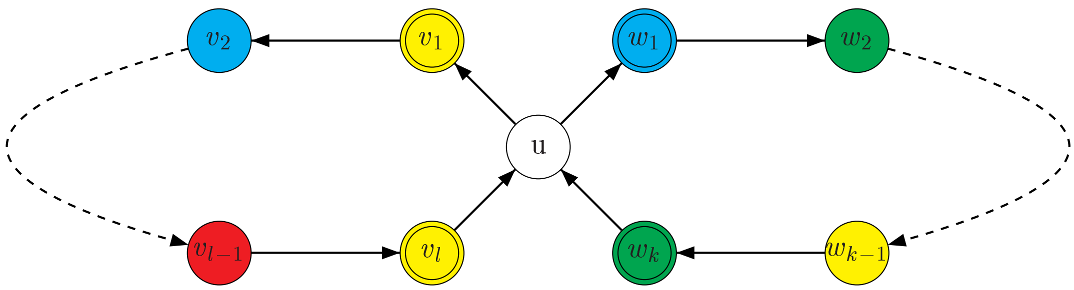
(2) A memoryless *strategy for player i is a mapping $\sigma : V_i \longmapsto V_{1-i}$.*

For Muller winning conditions, a player can have a w.s. but no winning memoryless strategy:

Definition 42 *Let $\mathcal{F}_i \subseteq (\mathcal{P}(C) \setminus \{\emptyset\})$, we call a* split *any pair of sets $X_1, X_2 \in \mathcal{F}_i$ such that $X_1 \cup X_2 \notin \mathcal{F}_i$.*

Proposition 43 *If in a Muller condition $(\mathcal{F}_0, \mathcal{F}_1)$ the set $\mathcal{F}_{1-i}$ has a split, then there exists a game $G, (\mathcal{F}_0, \mathcal{F}_1)$ such that player i has a w.s. but no memoryless w.s..*

Proof of proposition 43: Assume that $X_1, X_2 \in \mathcal{F}_{1-i}$ but $X_1 \cup X_2 \in \mathcal{F}_i$. Consider the following graph where nodes with one circle are vertices for 0 and nodes with two circles are vertices for 1. Color the vertices in such a way that $X_1 = \{\phi(v_j) : 0 < j \leq l\}$ and $X_2 = \{\phi(w_j) : 0 < j \leq k\}$. The vertex u is the only one left uncolored. Starting from anywhere, player i has a w.s. which consists, when in u, in alternatively visiting v_1 and w_1. But this player does not have any memoryless winning strategy.



$\dashv$ 43

Notation 44 *Given an arena G and Muller conditions $(\mathcal{F}_0, \mathcal{F}_1)$ we write W^i to denote the set of winning positions for player i.*

Theorem 45 *Let $(\mathcal{F}_0, \mathcal{F}_1)$ be Muller conditions,*

$$\text{player } i \text{ has a memoryless w.s. on all games with Muller conditions } (\mathcal{F}_0, \mathcal{F}_1)$$
$$\Longleftrightarrow$$
$$\mathcal{F}_{1-i} \text{ is closed under union.}$$

Notation 46 *Let $U \subseteq V$, be a set of vertices of $G = (V_0, V_1, E, C, \phi)$,*

(1) we write $G[U]$ to denote the partially colored subgraph of G induced by U:

$$G[U] = (V_0 \cap U, V_1 \cap U, E \cap (U \times U), C, \phi \restriction_U).$$

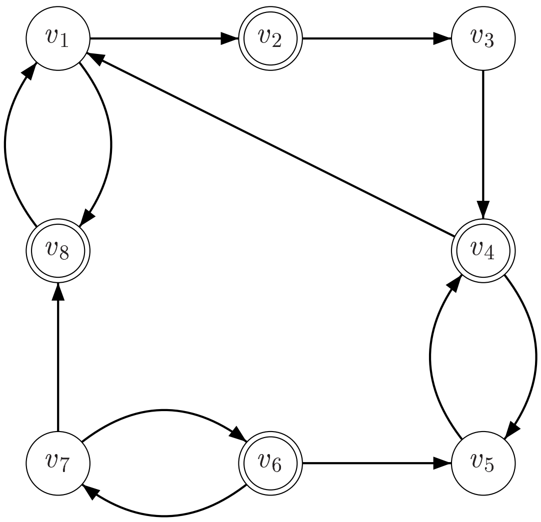
We say that $G[U]$ is a subarena *of G if it is an arena.*

*(2) Given $i \in \{0, 1\}$, a i-*trap *in G is any non empty set $U \subseteq V_0 \cup V_1$ such that*

$$\forall v \in U \cap V_i \quad vE \subseteq U \qquad \text{and} \qquad \forall v \in U \cap V_{1-i} \quad vE \cap U \neq \emptyset.$$

Clearly, a *i-trap U* induces a *subarena $G[U]$*. The idea of a *i-trap U* is that player $1 - i$ has a w.s. in order to force the play to remain inside U.

Example 47 In the arena G below, the set $\{v_1, v_8\}$ is a 1-trap, while the sets $\{v_1, v_2, v_3, v_4, v_8\}$ and $\{v_4, v_5\}$ are 0-traps. On the contrary, $\{v_4, v_5, v_6, v_7\}$ induces a subarena in G but is neither a 0-trap nor a 1-trap.



Lemma 48 *Let $G = (V_0, V_1, E, C, \phi)$ be an arena and $X \subseteq V$ be a i-trap in G and $\emptyset \neq Y \subseteq X$,*

$$Y \text{ is a } i\text{-trap in } G \iff Y \text{ is a } i\text{-trap in } G[X].$$

Proof of lemma 48: straightforward.

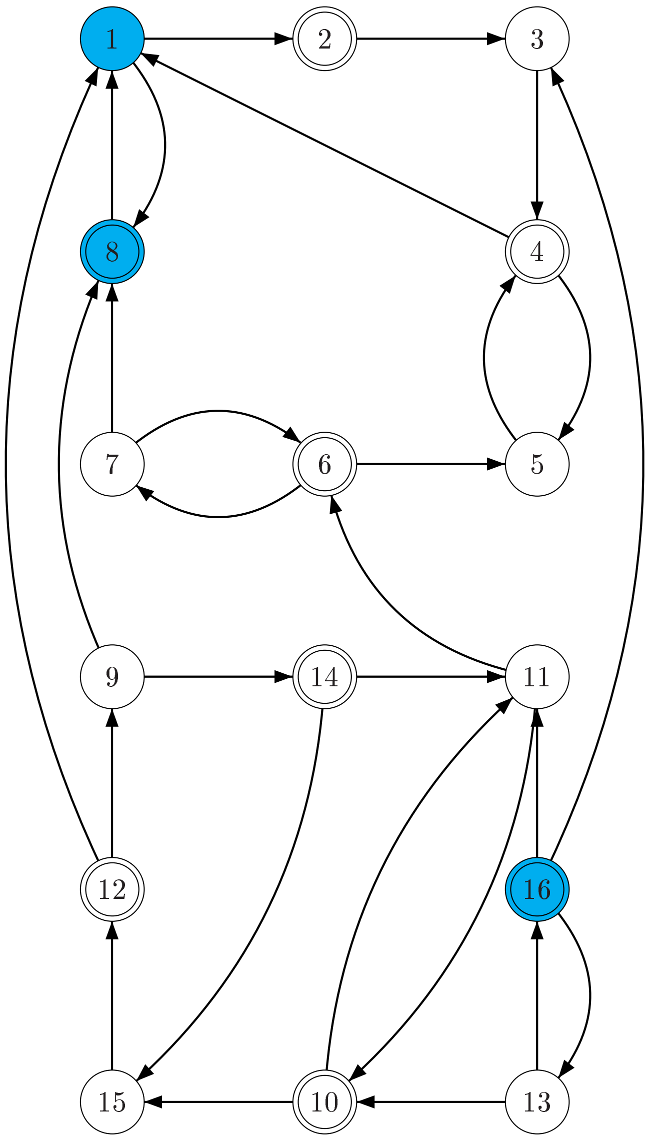
⊣ 48

Remark that this result fails for nested traps of different types: if X is a i-trap in G and Y is a $1-i$-trap in $G[X]$, then in general Y is neither a i-trap nor a $1-i$-trap in G.

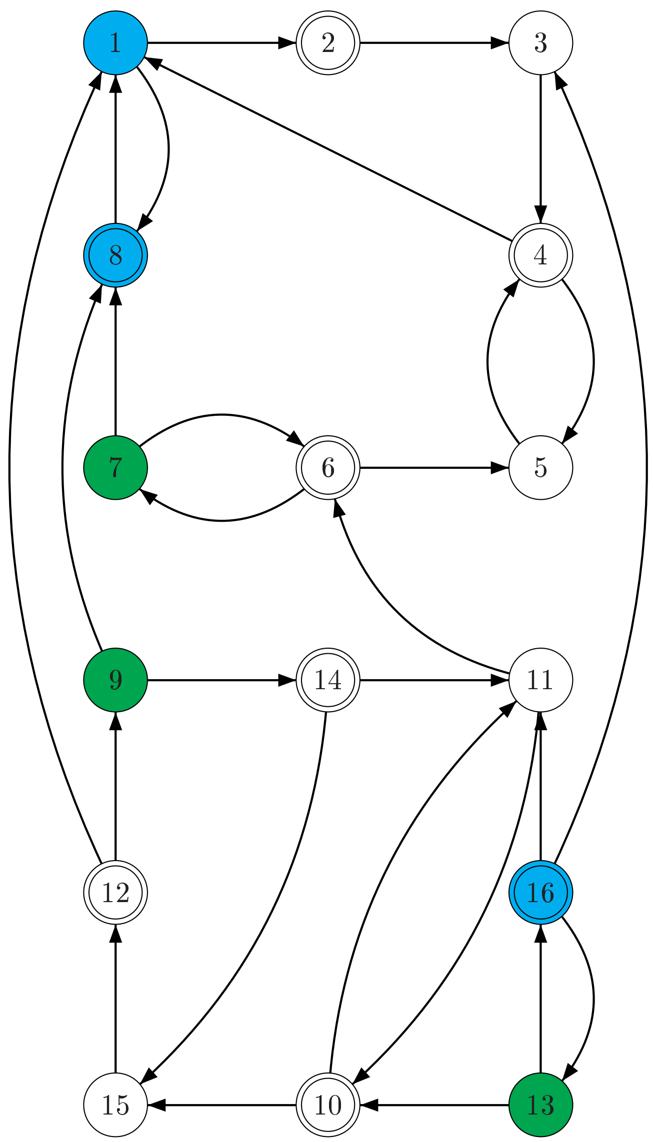
Definition 49 *Let $G = (V_0, V_1, E, C, \phi)$ be an arena and $X \subseteq V$,*
we define $Attr^i(G, X) \subseteq V$ as the greatest set of vertices such that player i has a winning strategy to attract the token from any vertex of $Attr^i(G, X)$ into X

- $X_0 = X$,
- $X_n + 1 = X_n \cup \{v \in V_i : \ vE \cap X_n \neq \emptyset\} \cup \{v \in V_{1-i} : \ vE \subseteq X_n\}$,
- $Attr^i(G, X) = \bigcup_{i \in \mathbb{N}} X_n$.

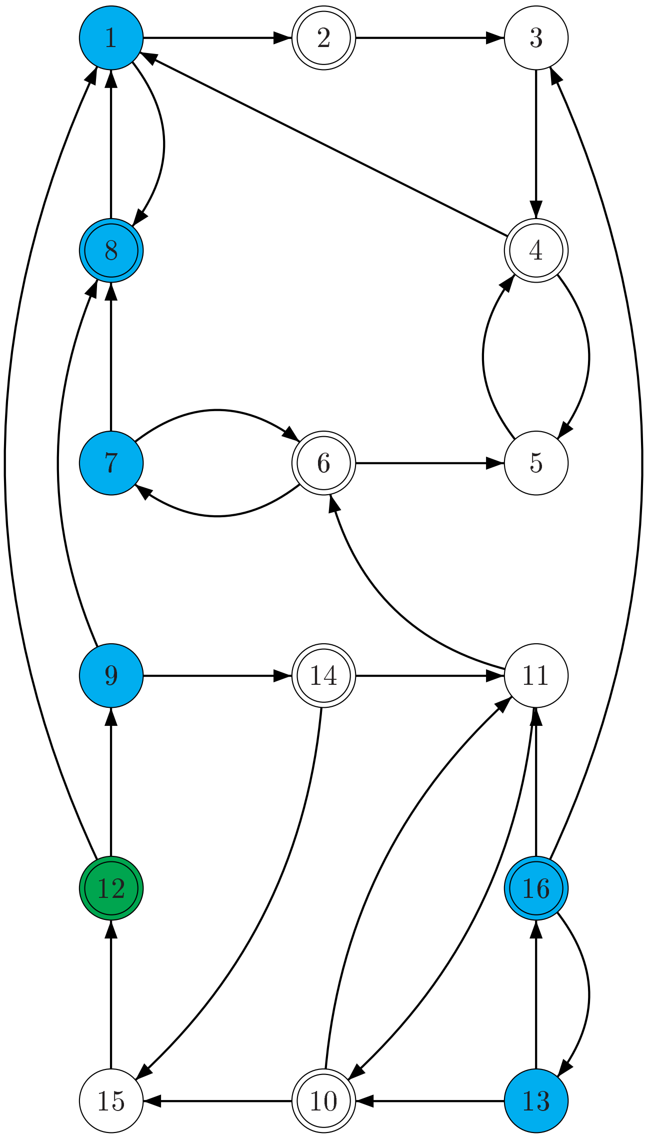
Example 50 We compute $\mathrm{Attr}^0(G, X)$ where G is the arena below and $X = \{1, 8, 16\}$ the set of blue states:



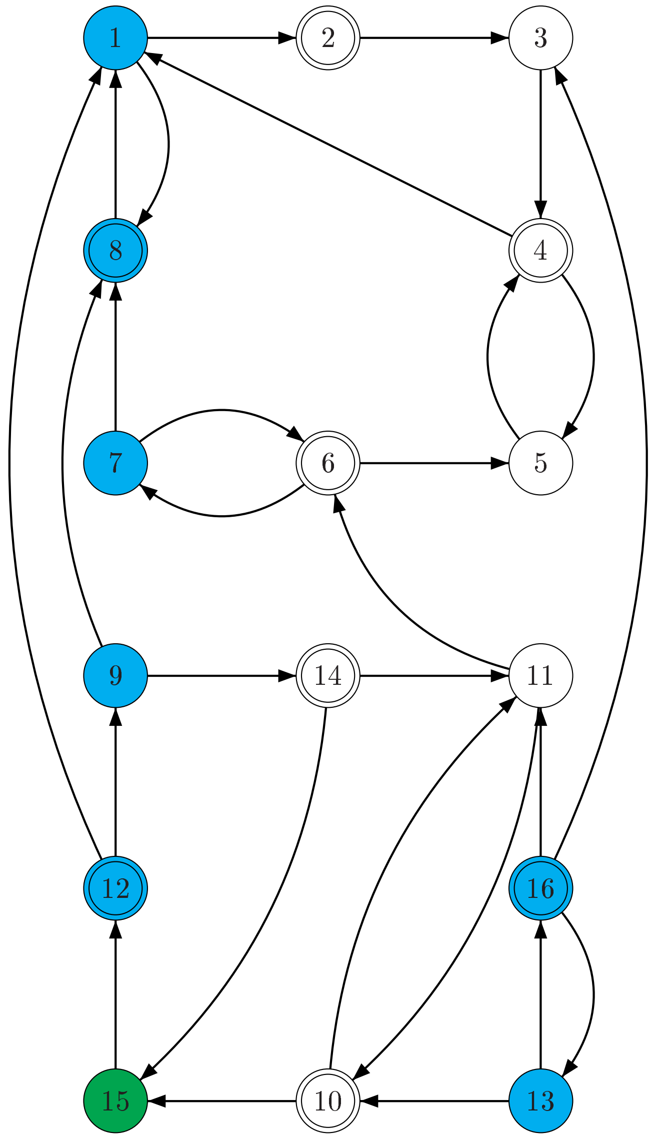
X_1 :



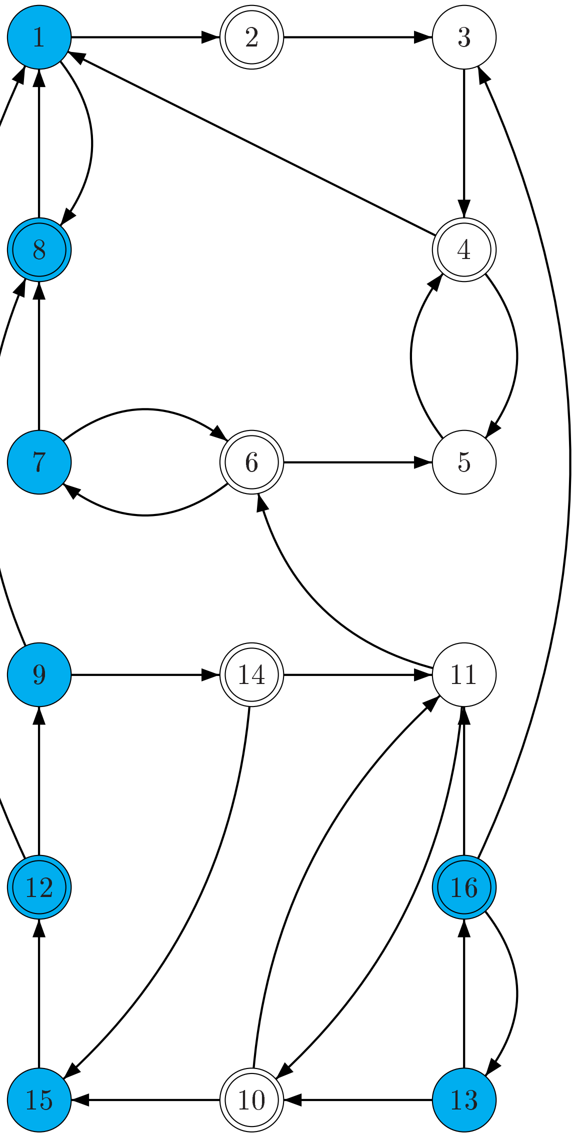
X_2 :



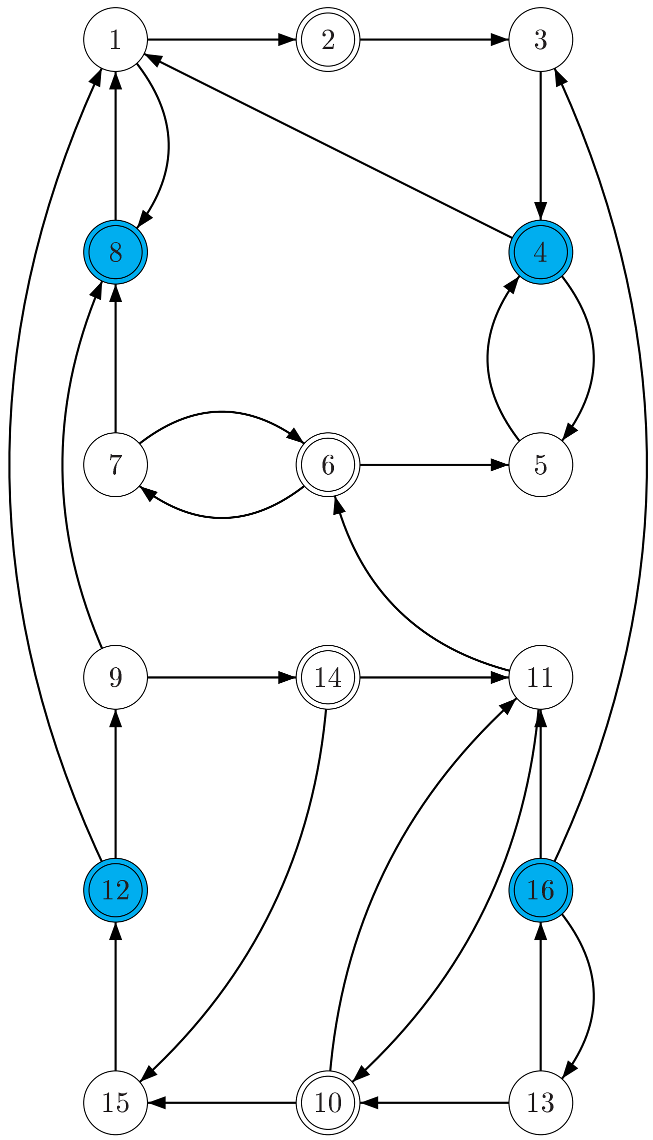
X_3 :

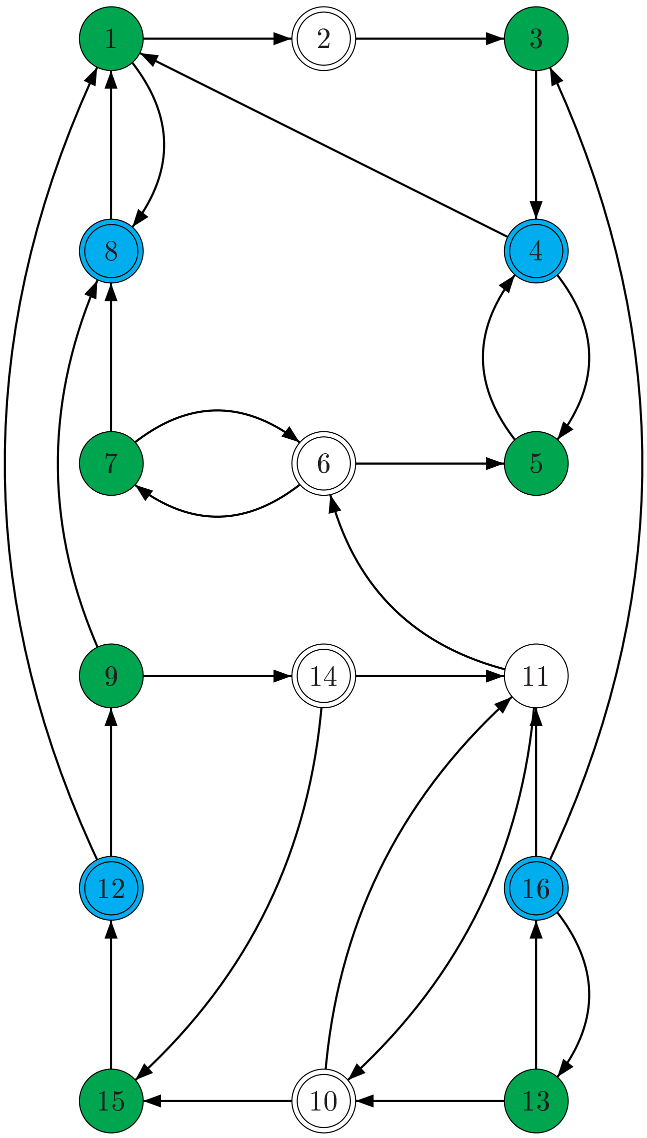


$X_4 = X_5 = \mathrm{Attr}^0(G, X)$:

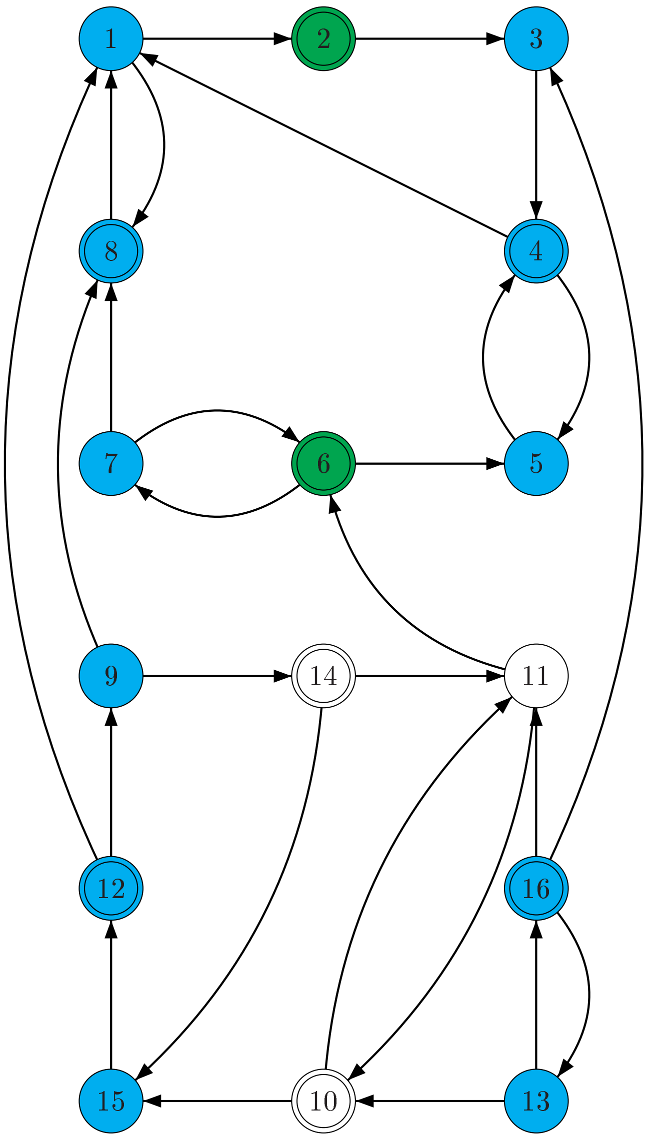


Example 51 We compute $\mathrm{Attr}^0(G, X)$ where G is the arena below and $X = \{4, 8, 12, 16\}$ the set of blue states:

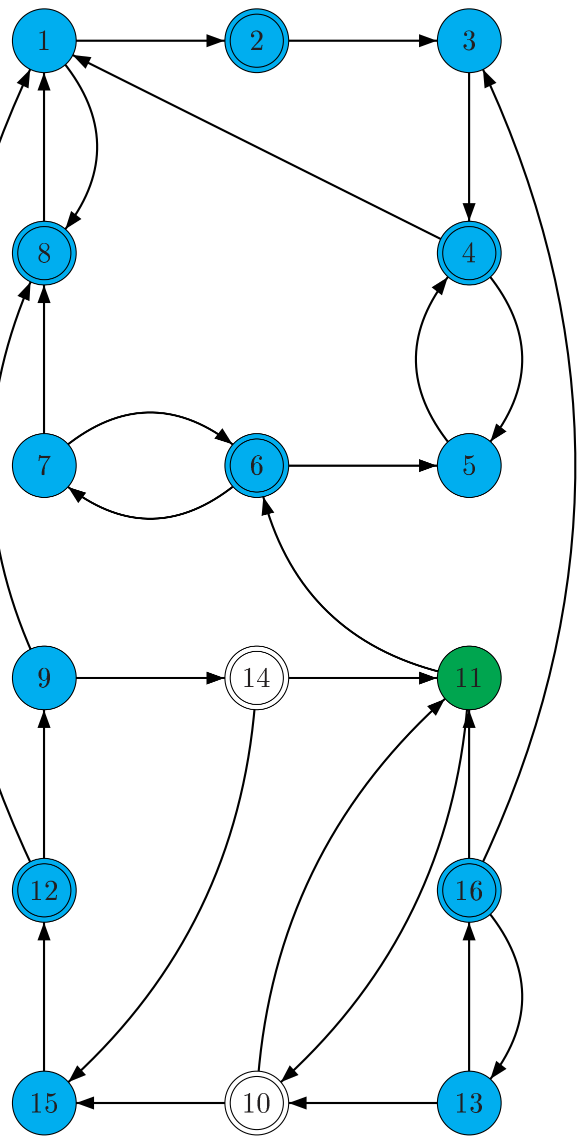



1
2
3
8
4
7
6
5
9
14
11
12
16
15
10
13

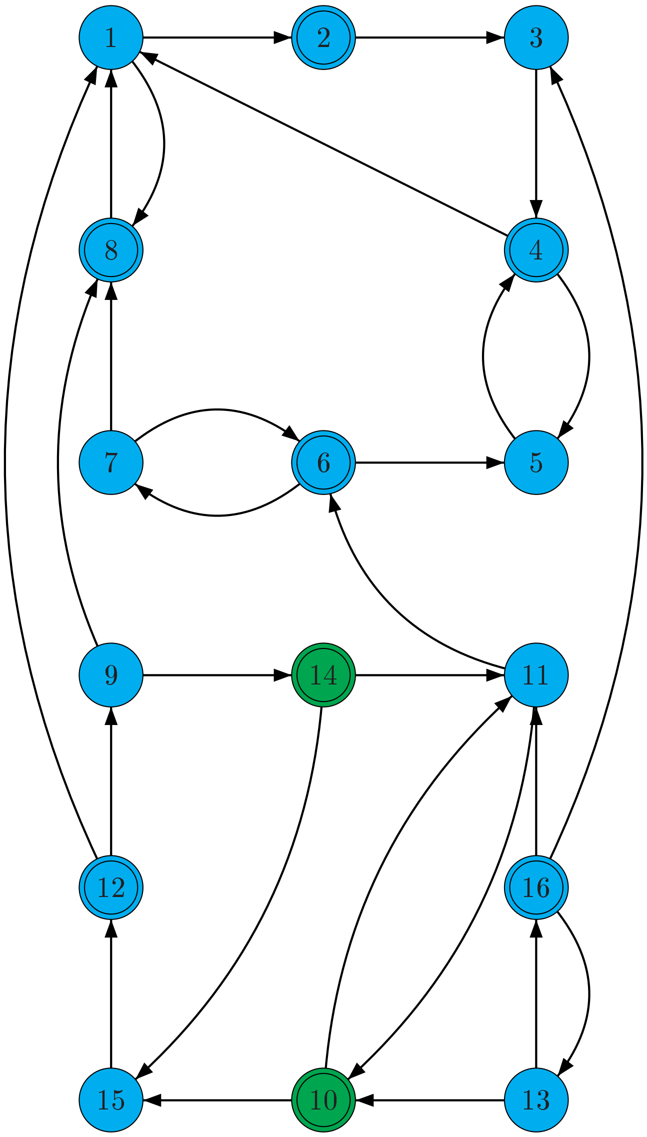
$X_2 = \{4, 8, 12, 16, 1, 3, 5, 7, 9, 13, 15\} \cup \{2, 6\}$



$X_3 = \{4, 8, 12, 16, 1, 3, 5, 7, 9, 13, 15, 2, 6\} \cup \{11\}$



$X_4 = \{4, 8, 12, 16, 1, 3, 5, 7, 9, 13, 15, 2, 6, 11\} \cup \{10, 14\} = X_5 = \mathrm{Attr}^0(G, X)$



Lemma 52 *Let $G = (V_0, V_1, E, C, \phi)$ be an arena and $X \subseteq V$,*

$$V \setminus Attr^i(G, X) \text{ is a } i\text{-trap in } G.$$

Proof of lemma 52: For any $v \in \mathrm{Attr}^i(G, X)$, let $rank(v)$ denote the smallest integer such that $v \in X_{rank(v)}$. Given any i vertex v, if V has a successor w in $\mathrm{Attr}^i(G, X)$, then $v \in X_{rank(w)+1}$, hence $v \in \mathrm{Attr}^i(G, X)$. If v is a $1 - i$ vertex with all its successors inside $\mathrm{Attr}^i(G, X)$, let $n = max\{rank(w) : w \in vE\}$, one obtains $v \in X_{n+1}$, which proves $v \in \mathrm{Attr}^i(G, X)$.

$\dashv$ 52

Lemma 53 *Let $G = (V_0, V_1, E, C, \phi)$ be an arena and $X \subseteq V$ be a i-trap in G,*

$$Attr^{1-i}(G, X) \text{ is a } i\text{-trap in } G.$$

Proof of lemma 53: Instead of verifying that the conditions defining a i-trap hold for $\mathrm{Attr}^{1-i}(G, X)$, we better note that player 1 has a w.s. to keep the token forever inside $\mathrm{Attr}^{1-i}(G, X)$: attract first the token into X, then always choose successors in X.

$\dashv$ 53

4.2.1 Parity Games

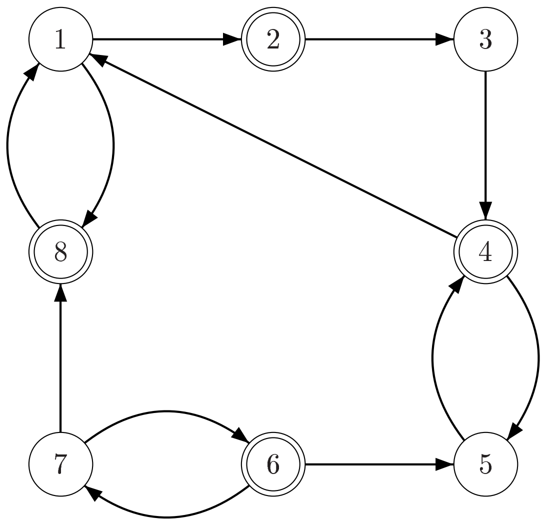
Definition 54 *Let* $G = (V_0, V_1, E, C, \phi)$ *be an arena and* $(\mathcal{F}_0, \mathcal{F}_1)$ *be Muller conditions,* $G, (\mathcal{F}_0, \mathcal{F}_1)$ *is a* **Parity game** *if*

- *colors are integers (i.e.* $C \subseteq \mathbb{N}$*)*
- *for any run* ρ, $\phi(Inf(\rho)) \in \mathcal{F}_0 \longleftrightarrow max\ \phi(Inf(\rho))$ *is even*

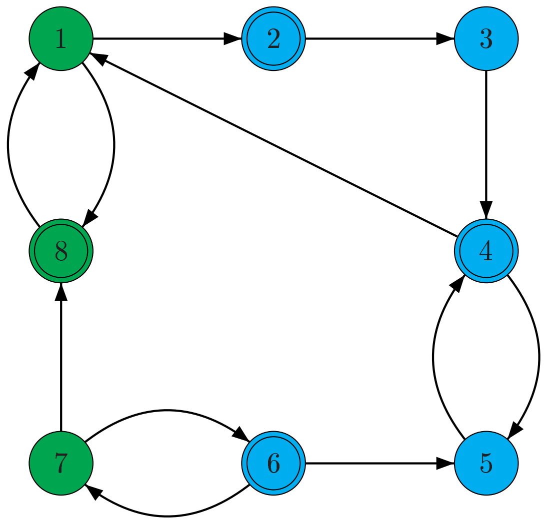
In other words, given any run ρ*, player* $i = max\ \phi(Inf(\rho)) \mod (2)$ *wins.*

Clearly these games are determined for they are just special cases of games played on Automata with Muller acceptance conditions.

Example 55 Numbers inside vertices indicate the colors.



In green are the vertices from which player 0 has a w.s., in blue, the ones where 1 has a w.s.:

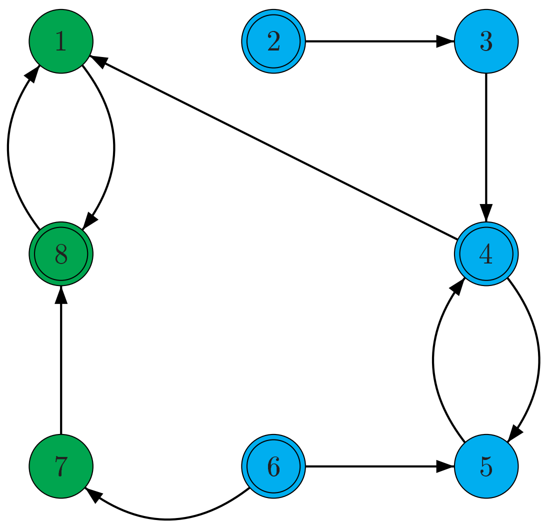


Theorem 56 *Let $G = (V_0, V_1, E, C, \phi)$ be an arena for the parity game ($C \subseteq \mathbb{N}$),*

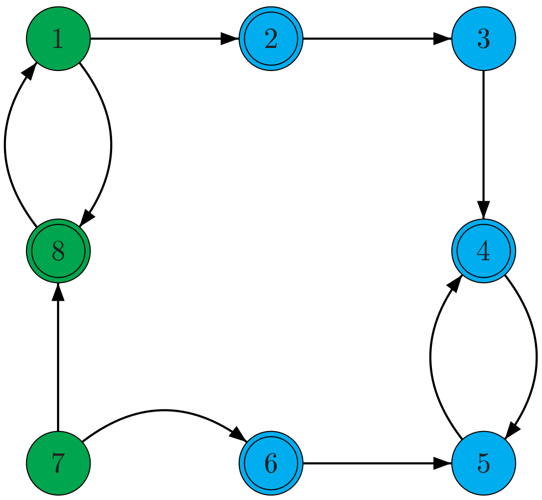
(1) W^i is a $1 - i$-trap in G,

(2) player i has a memoryless *winning strategy σ_i on W^i.*

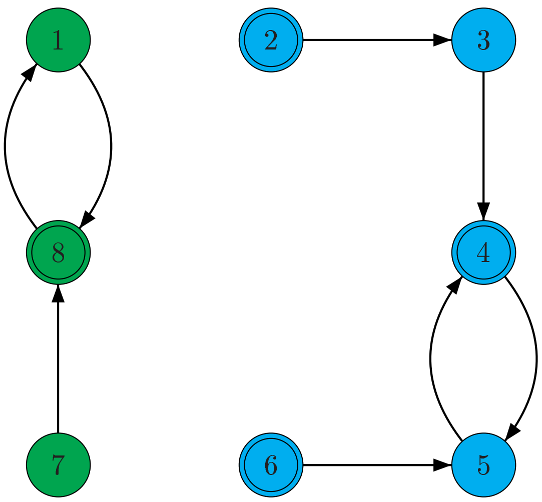
Example 57 Below we indicate the strategy followed by 0 on its winning set W^0:



Below we indicate the strategy followed by 1 on its winning set W^1:



Below are the two traps:



Proof of theorem 56: The proof goes by induction on $n =$ the maximal color of C.

If n=0 : the result is obvious since $W^0 = V$ and any strategy is winning for player 0.

If n>0 : we set $i = n \mod (2)$. By induction we construct sequences $(W_j^{1-i})_{0 \le j}$ and $(\sigma_j^{1-i})_{0 \le j}$ such that

(1) each $W_j^{1-i} \subseteq V$ is a i-trap in G and σ_j^i is a memoryless winning strategy for player $1 - i$ on $W_j^{1-i} \subseteq V$.

(2) If $h < j$, then $W_h^{1-i} \subsetneq W_j^{1-i}$ and σ_j^{1-i} extends σ_h^{1-i} from W_h^{1-i} to W_j^{1-i}.

We start with $W_0^{1-i} = \emptyset$. We assume W_j^{1-i} is defined, we show how to construct W_{j+1}^{1-i}:

We let $X_j = \mathrm{Attr}^{1-i}(G, W_j^{1-i})$. Since W_j^{1-i} is a i-trap, X_j is also a i-trap. And player $1-i$ has an obvious winning memoryless strategy $\tau_{X_j}^{1-i}$ on X_j: attract the token in a finite number of steps into W_j^{1-i} and then play according to the winning strategy σ_j^{1-i}.

We set $Y_j = V \smallsetminus X_j$. since Y_j is the complement of a $1-i$-attractor, it is a $1-i$-trap in G, in particular $G[Y_j]$ is a subarena of G. We consider the set $N_j = \{v \in Y_j : \; n = \phi(v)\}$ of all vertices of $G[Y_j]$ with maximum color, and $Z_j = Y_j \smallsetminus \mathrm{Attr}^i(G[Y_j], N_j)$. We remark:

(1) Z_j is a i-trap in the arena $G[Y_j]$ (since it is the complement of an i-attractor in this arena).

(2) no vertice of Z_j has color n.

We concentrate on the parity game played on $G[Z_j]$. By induction hypothesis, there exists a partition of Z_j in winning sets Z_j^0 and Z_j^1 for respectively player 0 and 1, with coresponding w.s. $\tau_{Z_j}^0$ and $\tau_{Z_j}^1$.

We set $W_{j+1}^{1-i} = X_j \cup Z_j^{1-i}$. The definition of the strategy σ_{j+1}^{1-i} on W_{j+1}^{1-i} is obvious: if the token is in X_j then player $1-i$ plays according to $\tau_{X_j}^{1-i}$, otherwise, if the token is in Z_j^{1-i} then he uses the strategy $\tau_{Z_j}^{1-i}$ given by the induction hypothesis.

We must still verify that the i-vertices in W_{J+1}^{1-i} have successors only inside W_{J+1}^{1-i}:

- i-vertices in X_j have successors only in X_j since X_j is a i-trap in G.
- i-vertices in Z_j^{1-i} cannot have successors in $\mathrm{Attr}^i(G[Y_j], N_j)$ (because the complement of a i-attractor is a i-trap),
- i-vertices in Z_j^{1-i} cannot have successors in Z_j^i (because Z_j^{1-i} is a i-trap in $G[Z_j]$),
- each $1-i$-vertex has at least one successor in W_{j+1}^{1-i}.

Hence W_{J+1}^{1-i} is a i-trap.

Now given a run $\rho = v_0 v_1 v_2 \dots$ that is consistent with the strategy σ_{j+1}^{1-i}:

- if the token hits at some point the set X_j, then from this very moment it will remain inside X_j and player $1-i$ will win the play by applying his winning strategy $\tau_{X_j}^{1-i}$ on X_j.
- if the token never hits X_j, then the whole run is consistent with the strategy $\tau_{Z_j}^{1-i}$ on Z_j and $1-i$ also wins the game.

Let j be the first integer such that $W_j^{1-i} = W_{j+1}^{1-i}$. We set $W^{1-i} = W_j^{1-i}$ and claim it is the winning set for player $1-i$ in the parity game on G, with $\sigma^{1-i} = \sigma_j^{1-i}$.

In order to define a memoryless winning strategy for player i on $W^i = V \smallsetminus W^{1-i}$ we remark that

$$W_j^{1-i} \subseteq X_j = \mathrm{Attr}^{1-i}(G, W_j^{1-i}) \subseteq W_{j+1}^{1-i} = W_j^{1-i}.$$

Hence $W^{1-i} = \mathrm{Attr}^{1-i}(G, W^{1-i})$, thus W^i, being the complement of an $1-i$-attractor, is a $1-i$-trap. We set

$$N = \{v \in W^i : \; n = \phi(v)\} \qquad \text{and} \qquad Z = W^i \smallsetminus \mathrm{Attr}^i(G[W^i], N).$$

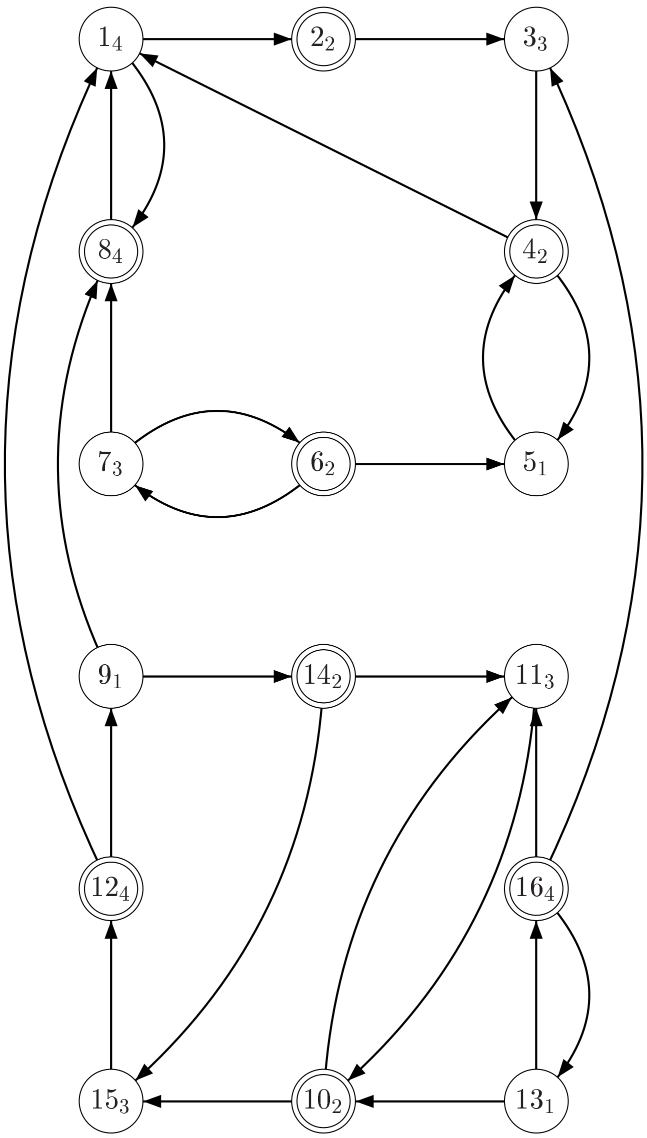
No vertex from $G[Z]$ has color n, hence one can solve the parity game on $G[Z]$ by induction hypothesis, but this time the winning set of player i is empty (otherwise it would have been added to W_j^{1-i}). Therefore, player i has a memoryless winning strategy τ_{Z^i} that allows him to win all over $G[Z]$. One finally defines the strategy σ^i for $v \in W^i \cap V_i$ by:

- if the token visits a vertex $v \in N$ (hence colored by the maximal color n), then player i moves it to any successor vertex inside his winning set W^i (there is always such a one since W^i is a $1-i$-trap). If
- if the token visits $\mathrm{Attr}^i(G[W^i], N) \setminus N$ then i attracts it in a finite number of moves to N. Hence, the token, after finitely many steps will keep on visiting the maximal color.
- if the token stands inside Z, then i plays his winning strategy τ_{Z^i} on Z

Clearly, σ^i is a winning strategy for i on $W^i \cap V_i$ since, in any play ρ consistent with σ^i: either the token visits infinitely often the maximal color (which is even), or, after some time, it stays forever inside Z, in which case a suffix from ρ is consistent with τ_{Z^i} which proves i wins.

$\dashv$ 56

Example 58 In the example below, colors are indicated as subscripts: node h_k has color k.



since the maximal color is 4 which is even, we are in the case $i = 0$.

(1) (a) $W_0^1 = \emptyset$

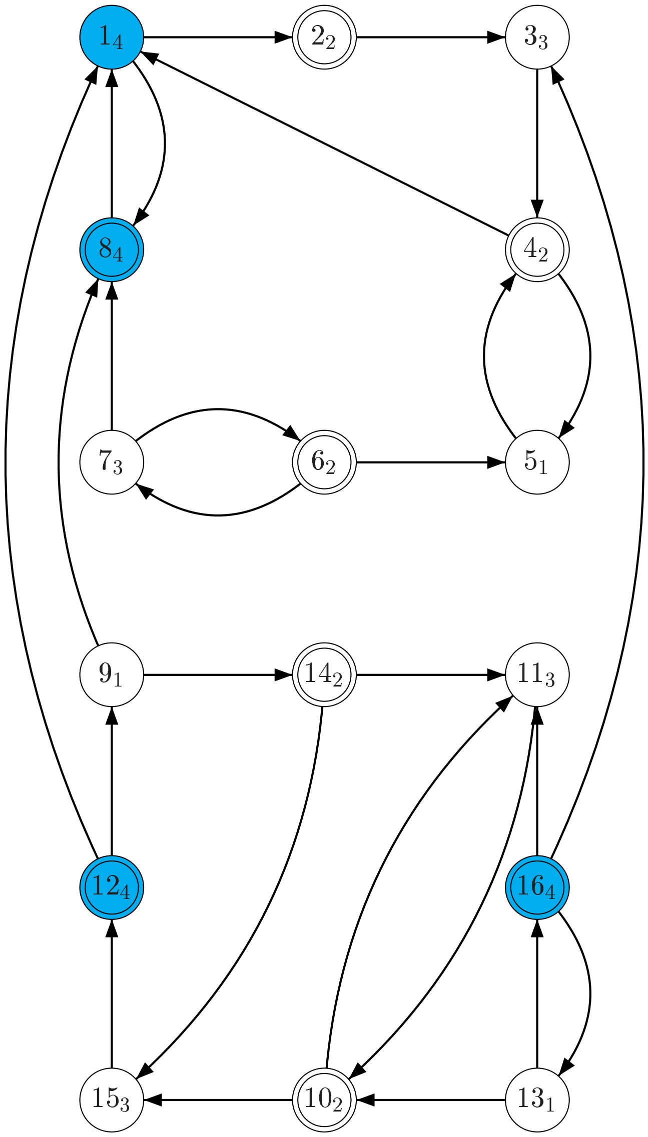
(b) $X_0 = \mathrm{Attr}^1(G, W_0^1) = \mathrm{Attr}^1(G, \emptyset) = \emptyset$

(c) $Y_0 = V \setminus X_0 = V \setminus \emptyset = V$

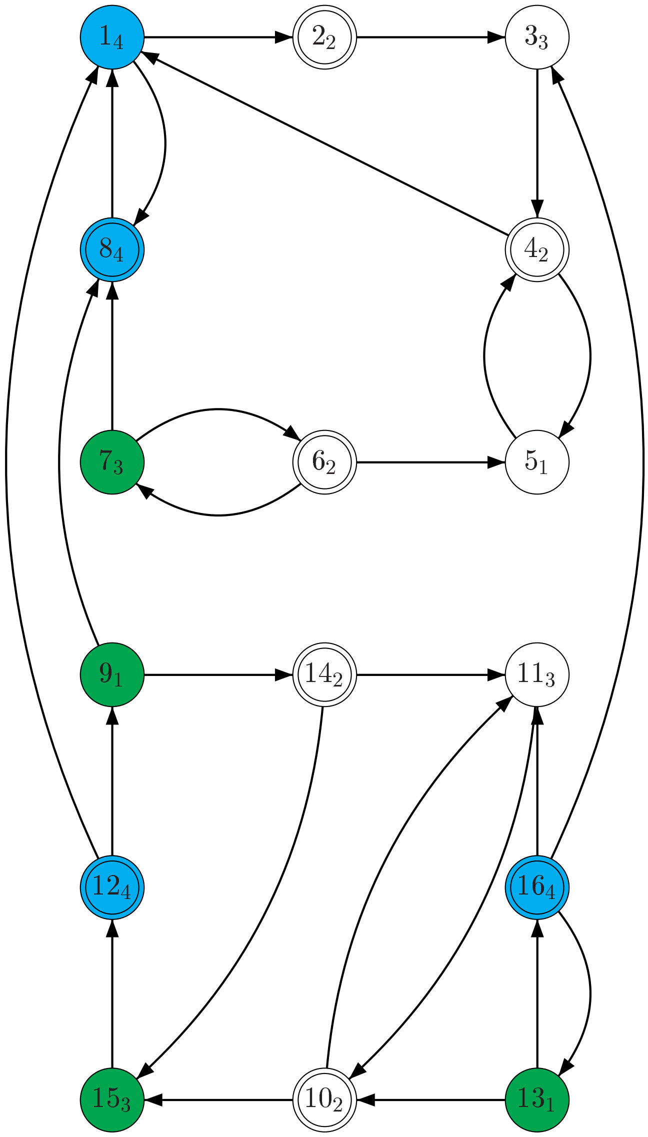
(d) $N_0 = \{v \in Y_0 : \ 4 = \phi(v)\} = \{v \in V : \ 4 = \phi(v)\} = \{4_4, 8_4, 12_4, 16_4\}$

(e) $Z_0 = Y_0 \setminus \mathrm{Attr}^0(G[Y_0], N_0) = Y_0 \setminus \mathrm{Attr}^0(G[V], N_0) = V \setminus \mathrm{Attr}^0(G, \{1_4, 8_4, 12_4, 16_4\})$
$= V \setminus$ least fixpoint of the sequence $(H_l)_{l \in \mathbb{N}}$

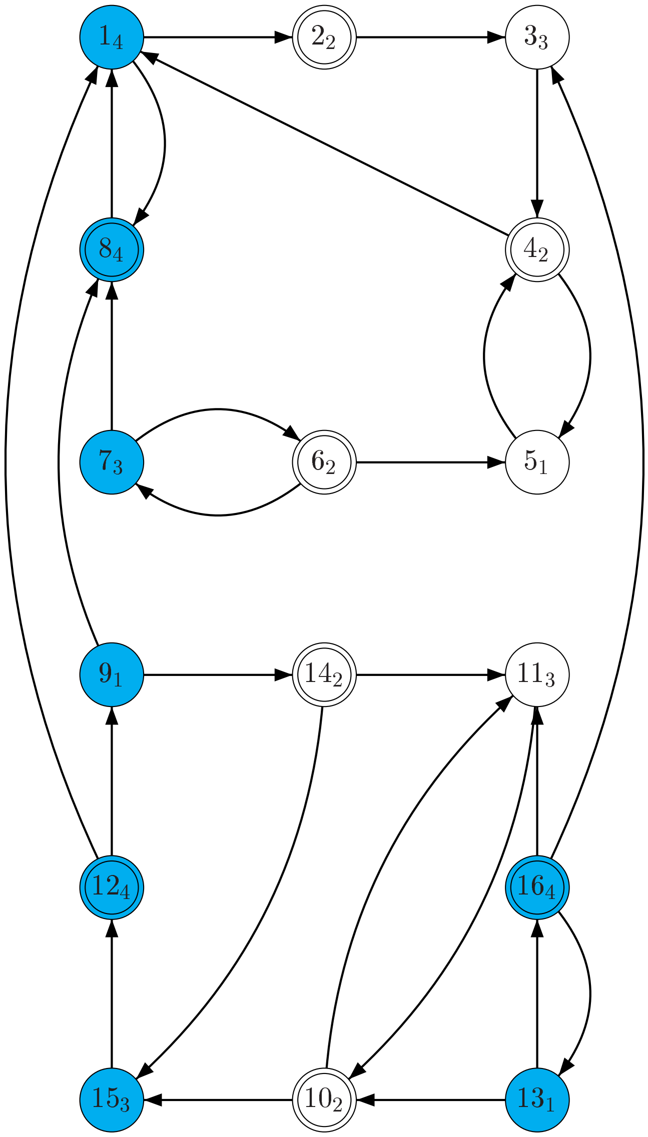
(1) $H_0 = \{1_4, 8_4, 12_4, 16_4\}$



(2) $H_1 = \{1_4, 8_4, 12_4, 16_4\} \cup \{7_3, 9_1, 13_1, 15_3\}$

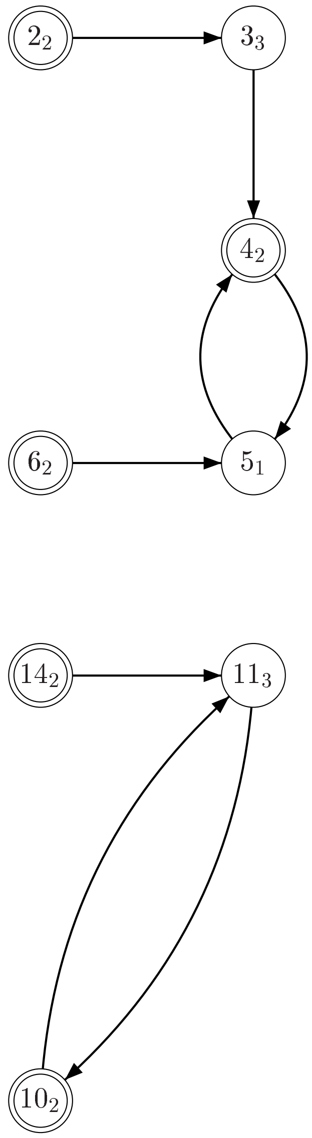


(3) $H_2 = \{4_4, 8_4, 12_4, 16_4, 1_1, 3_3, 5_1, 7_3, 9_1, 13_1, 15_3\} = H_1$

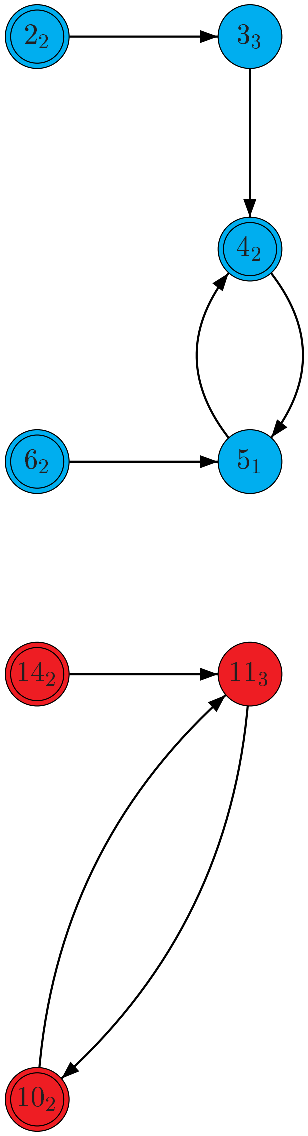


(f) ◦ Z_0 is a 0-trap in the arena $G[Y_0]$ (i.e. in G),
◦ nodes in Z_0 have color at most 3.

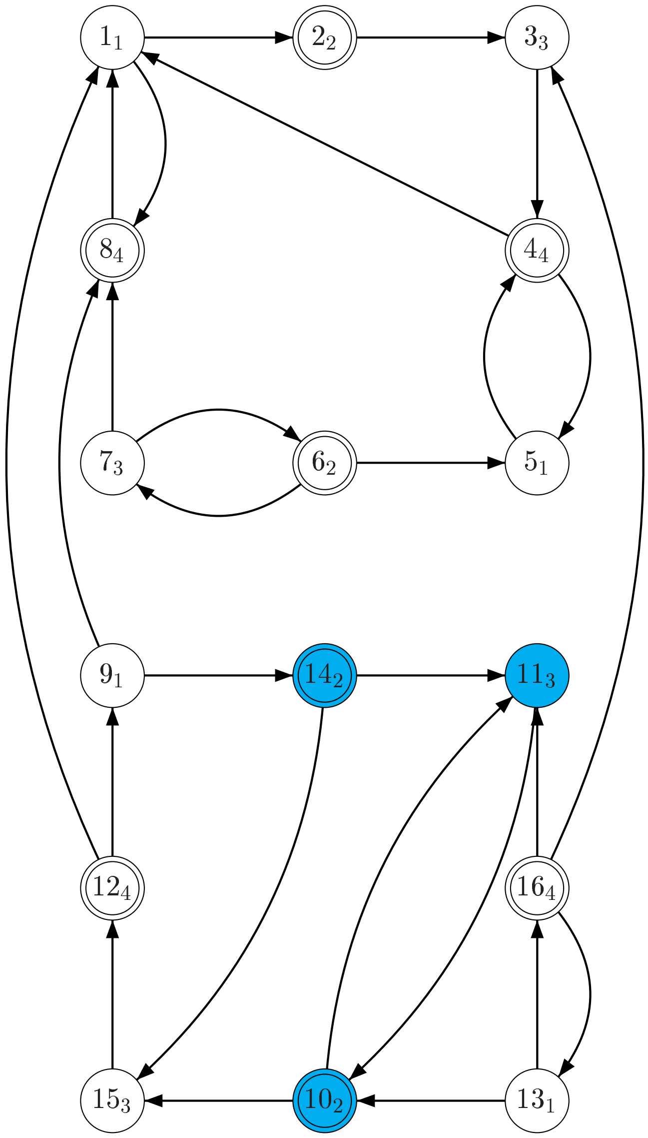
We concentrate on the parity game played on $G[Z_0]$.



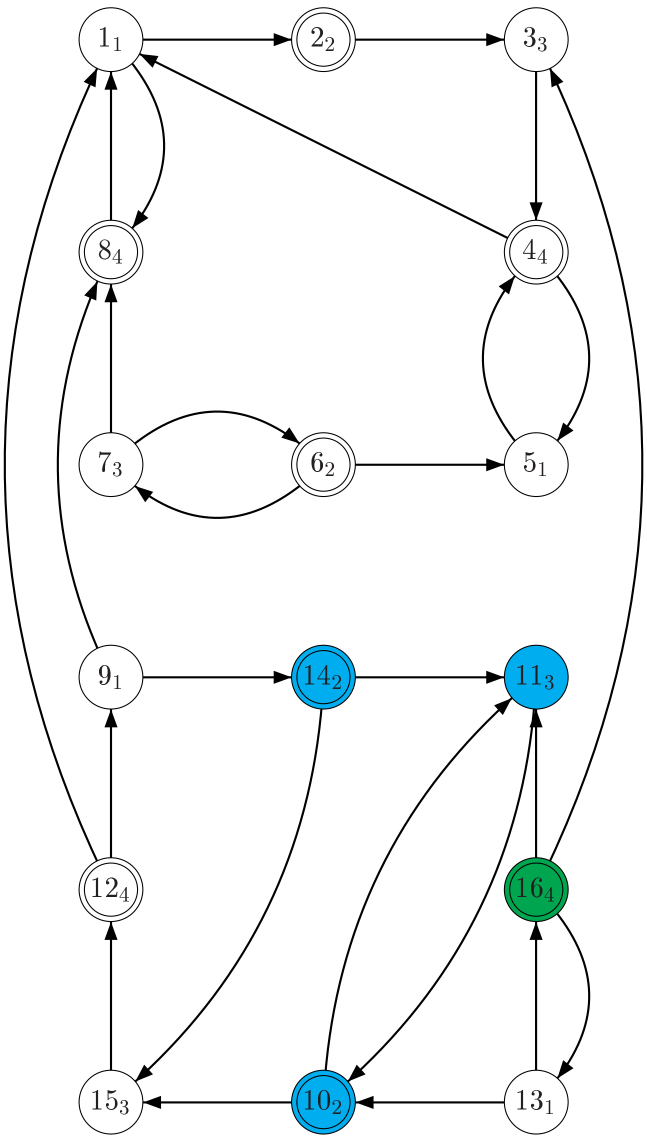
(g) Easily (without running the whole algorithm for Z_0) one gets Z_0^0 the set of winning vertices for 0 and Z_0^1 the set of winning vertices for 1, together with the coresponding winning strategies (obvious since there is no vertex with more than one successor):

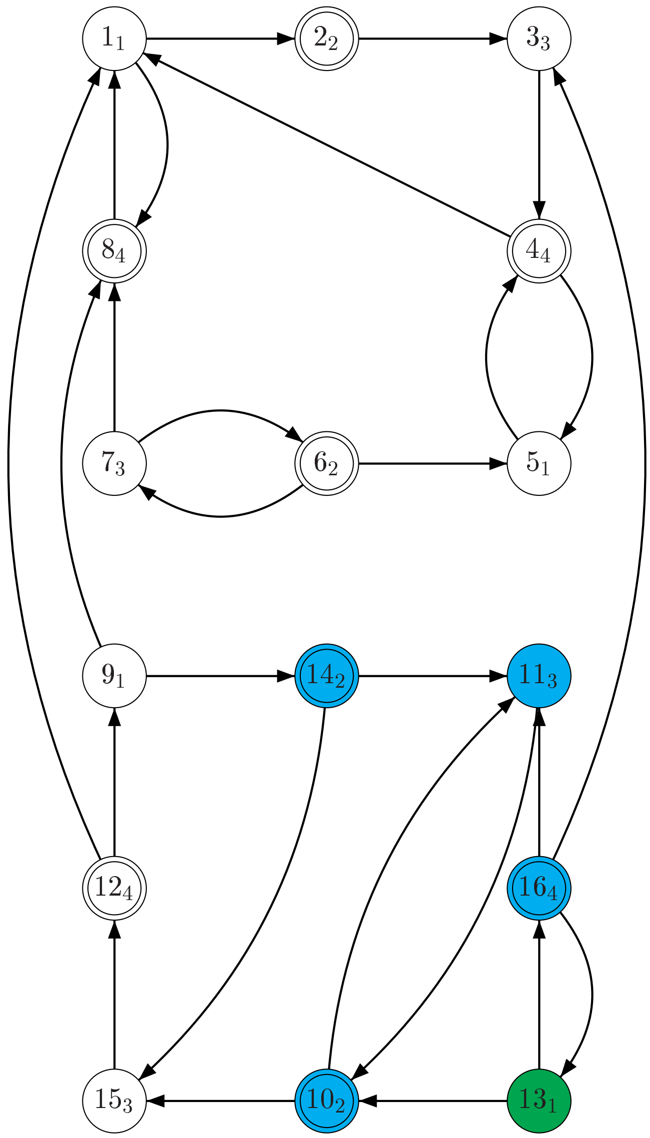


(2) (a) $W_1^1 = X_0 \cup Z_0^1 = \emptyset \cup \{10_2, 11_3, 14_2\} = \{10_2, 11_3, 14_2\}$, with the winning strategy for 1 being obvious on W_1^1



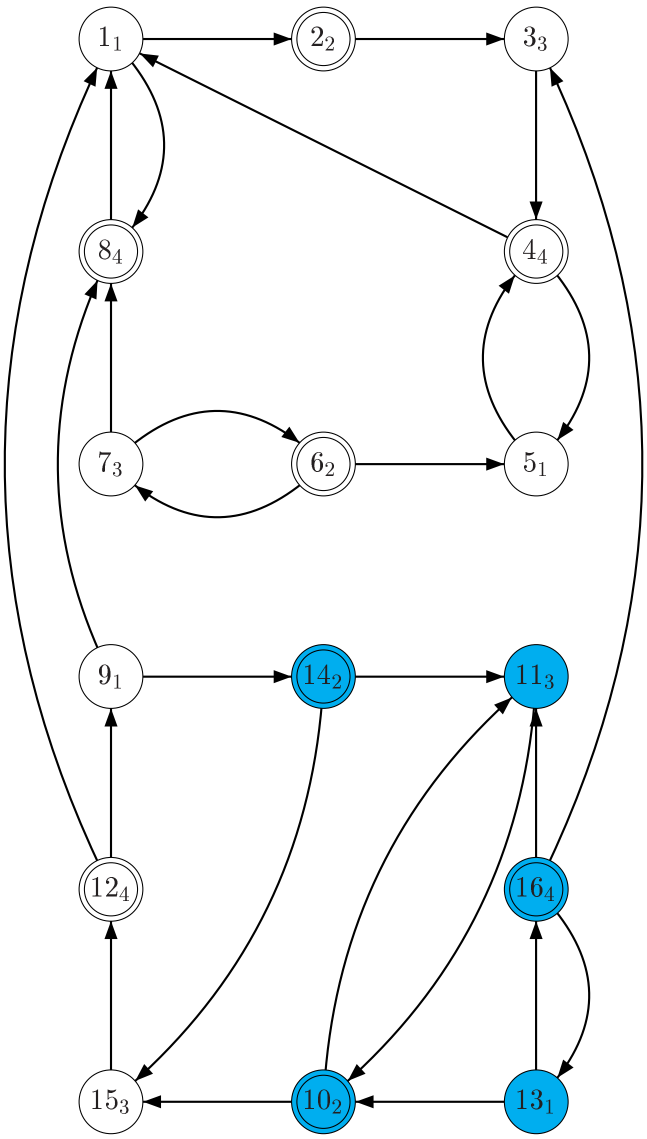
(b) $X_1 = \mathrm{Attr}^1(G, W_1^1) = \mathrm{Attr}^1(G, \{10_2, 11_3, 14_2\})$



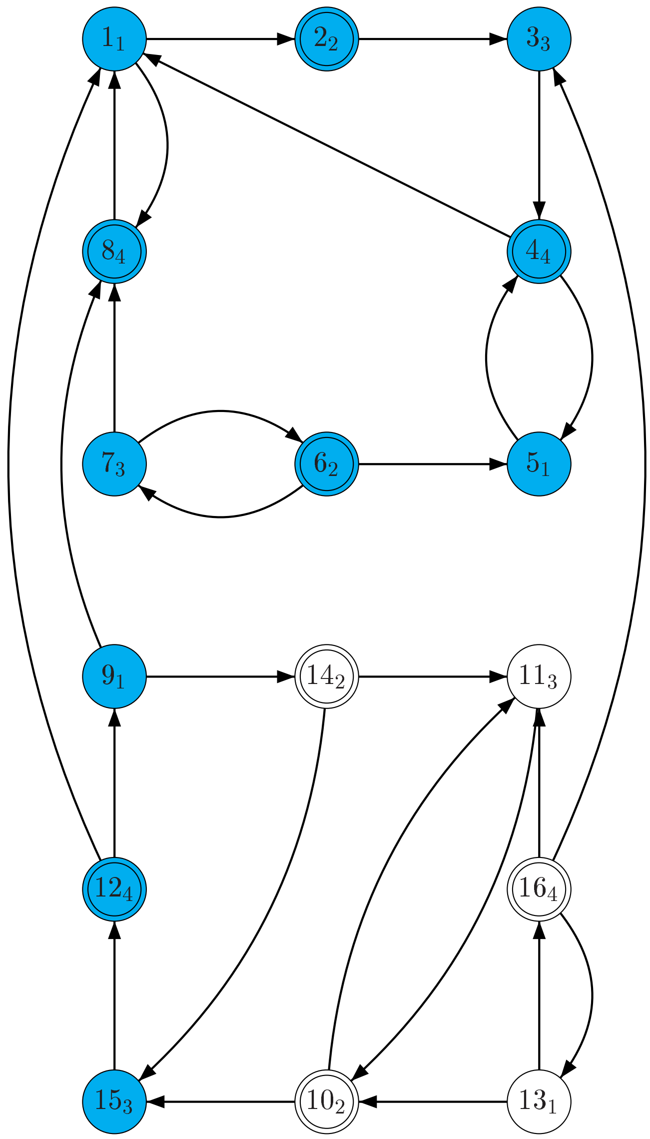


1_1
2_2
3_3
8_4
4_4
7_3
6_2
5_1
9_1
14_2
11_3
12_4
16_4
15_3
10_2
13_1

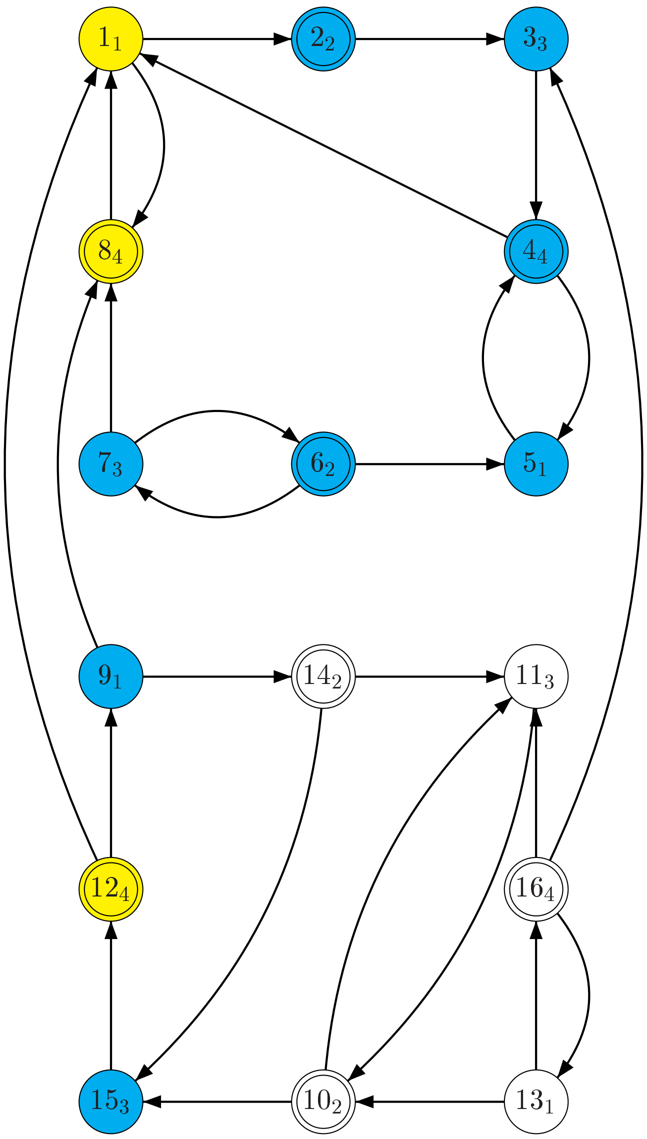
$X_1 = \{10_2, 11_3, 13_1, 14_2, 16_4\}$



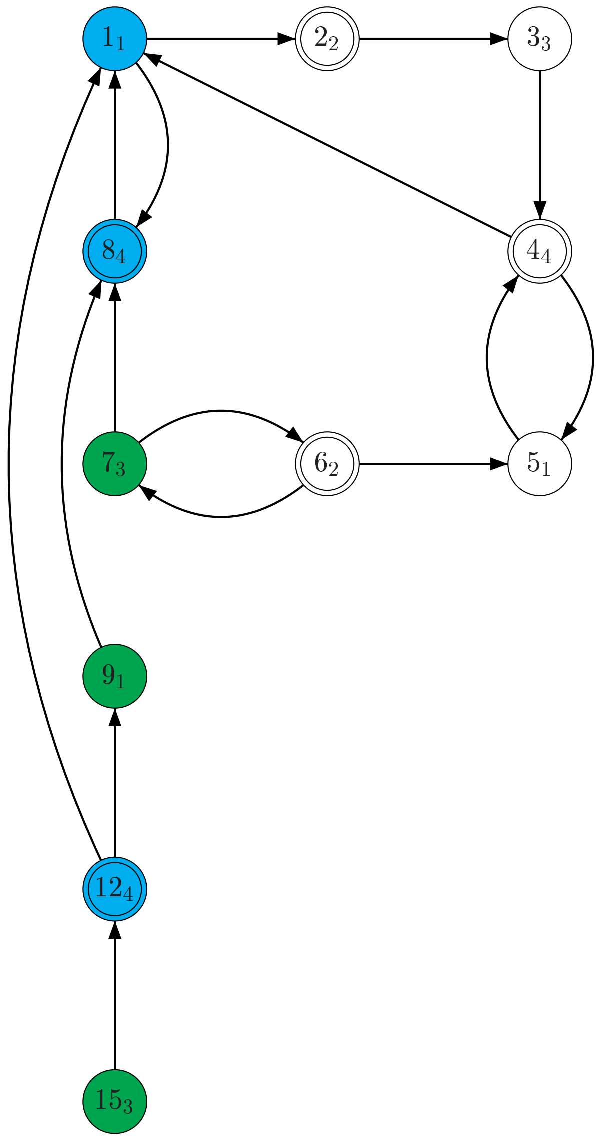
(c) $Y_1 = V \setminus X_1$



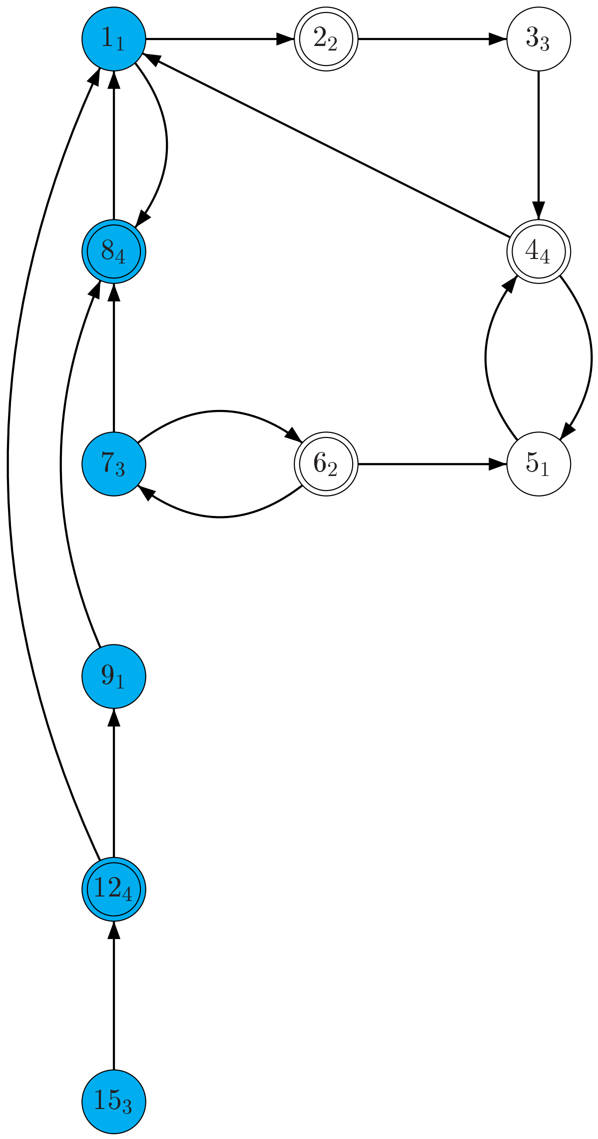
(d) $N_1 = \{v \in Y_1 : 4 = \phi(v)\} = \{1_4, 8_4, 12_4\}$



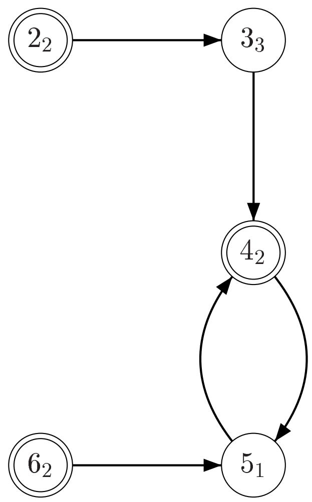
(e) $\mathrm{Attr}^0(G[Y_1], N_1)$



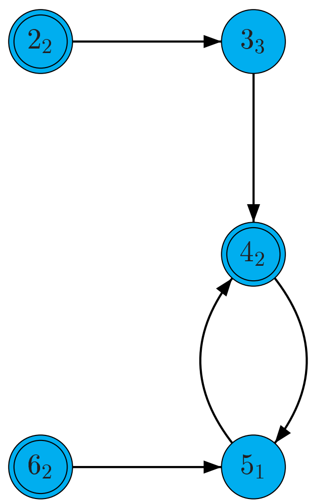
$\mathrm{Attr}^0(G[Y_1], N_1) =$



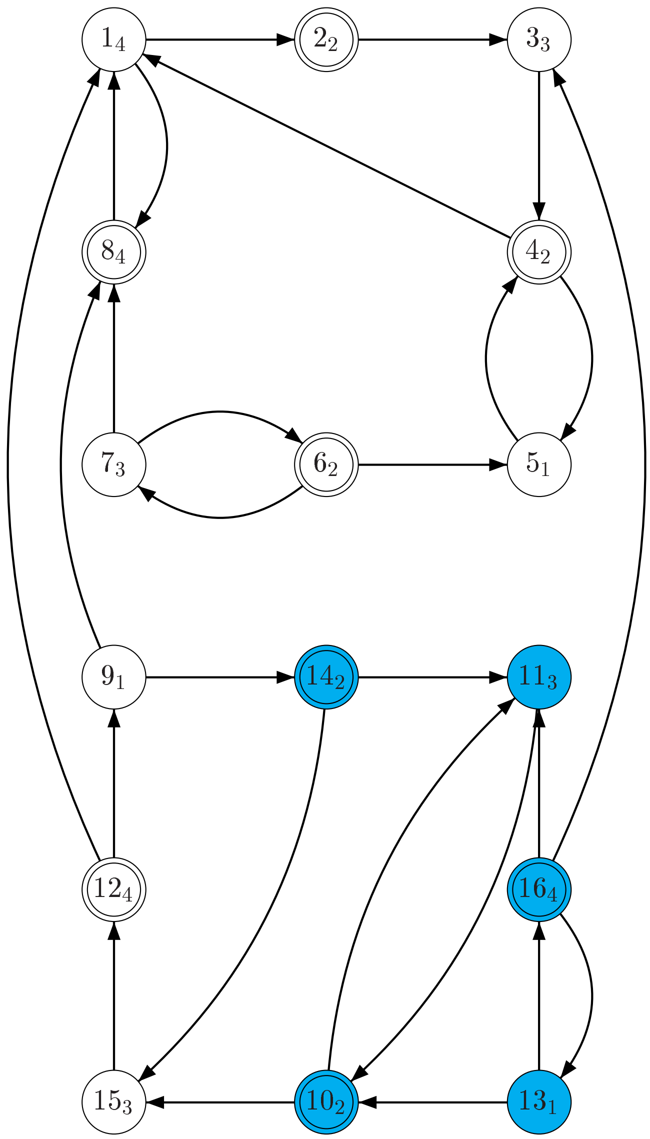
(f) $Z_1 = Y_1 \setminus \mathrm{Attr}^0(G[Y_1], N_1) = \{2_2, 3_3, 4_2, 5_1, 6_2\}$. We concentrate on the game played on $G[Z_1]$:



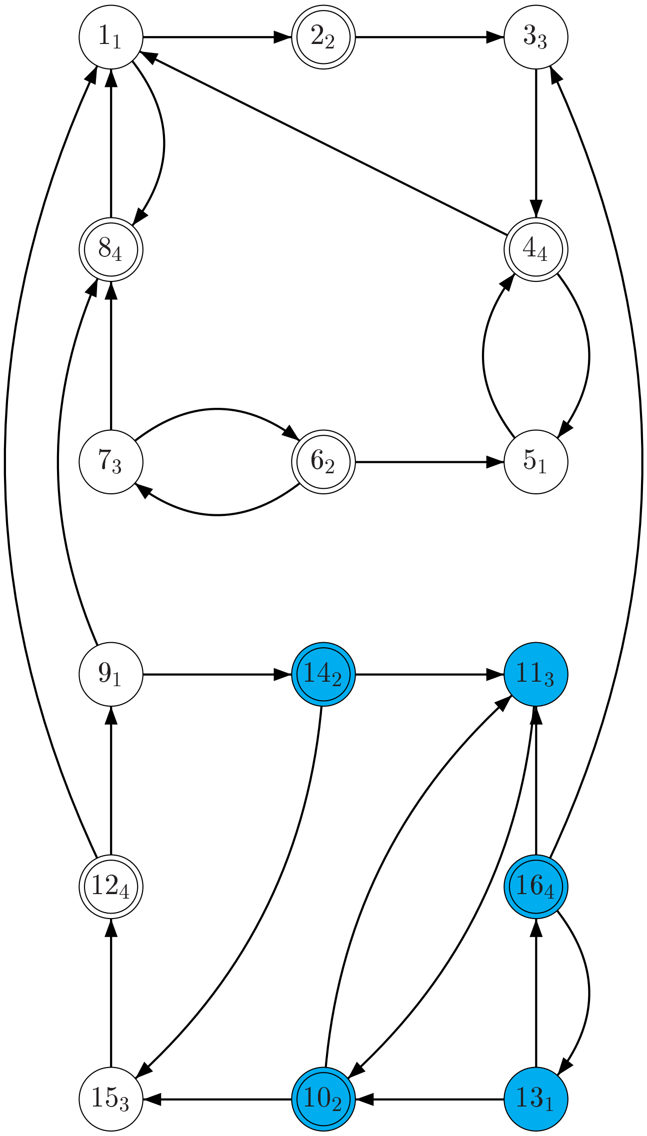
(g) Easily (without running the whole algorithm for Z_1) one gets Z_1^0 the set of winning vertices for 0, and the coresponding winning memoryless strategy. Clearly $Z_1^1 = \emptyset$



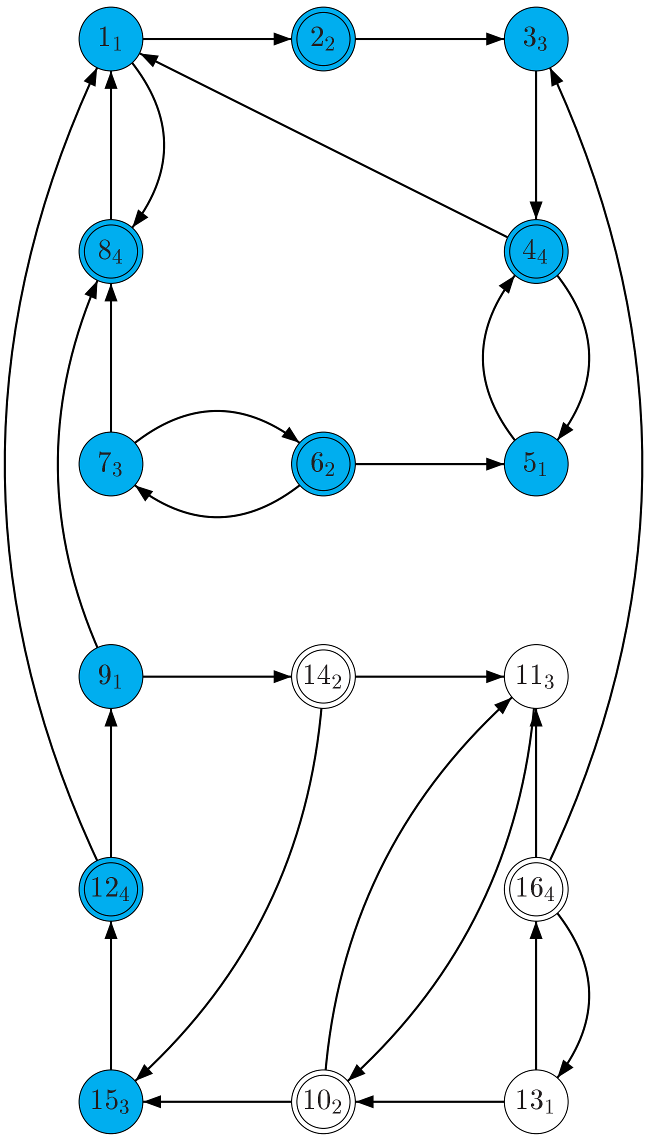
(3) (a) $W_2^1 = X_1 \cup Z_1^1 = \{10_2, 11_3, 13_1, 14_2, 16_4\} \cup \emptyset = \{10_2, 11_3, 13_1, 14_2, 16_4\}$.



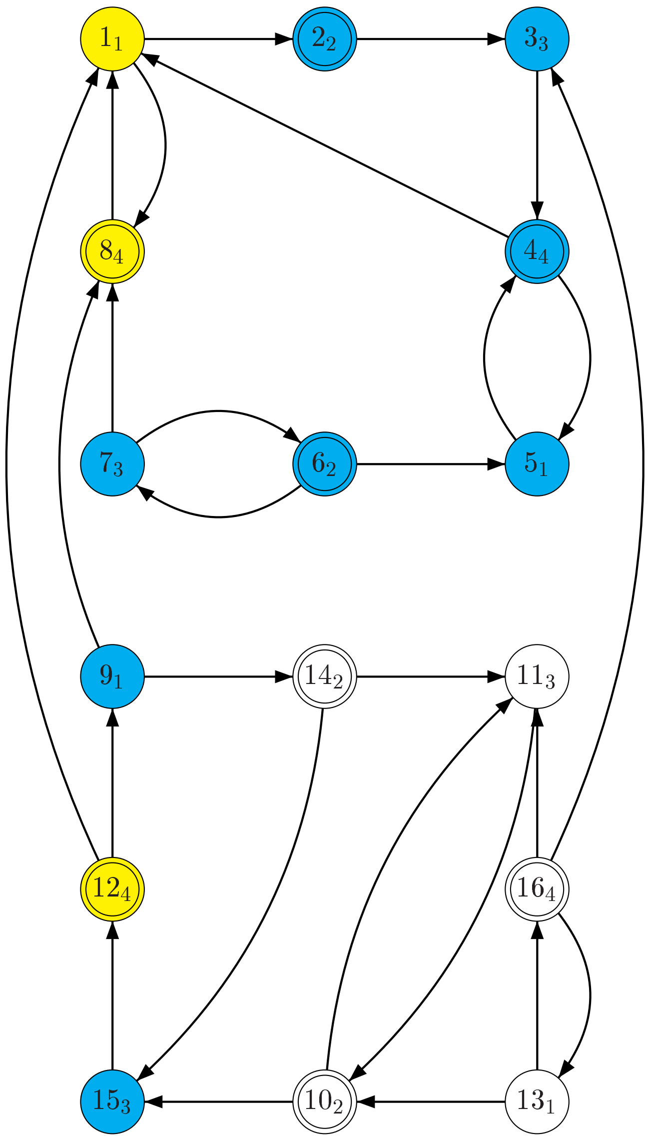
(b) $X_2 = \mathrm{Attr}^1(G, W_2^1) = \mathrm{Attr}^1(G, \{10_2, 11_3, 13_1, 14_2, 16_4\})$



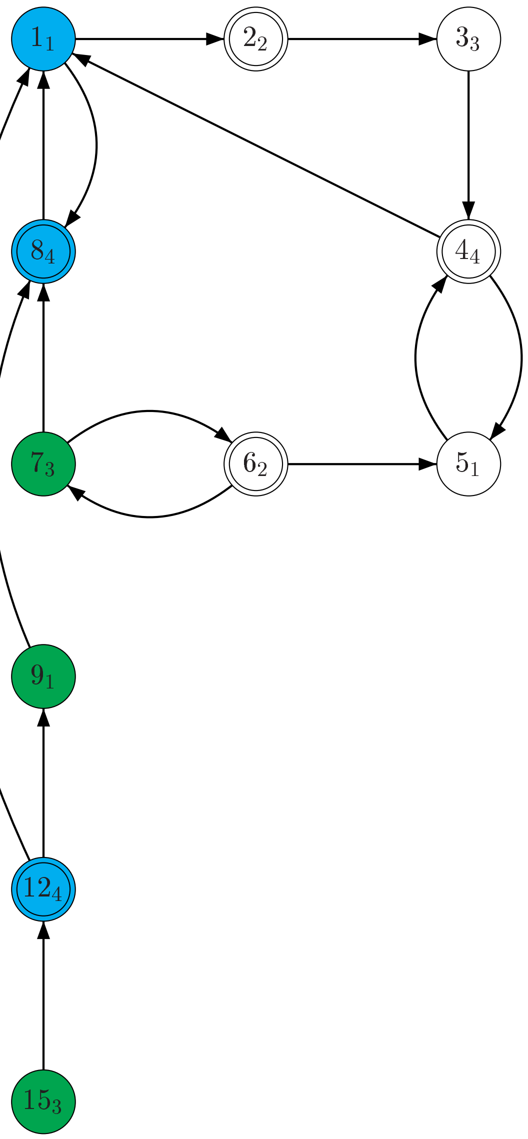
(c) $Y_2 = V \setminus X_2$



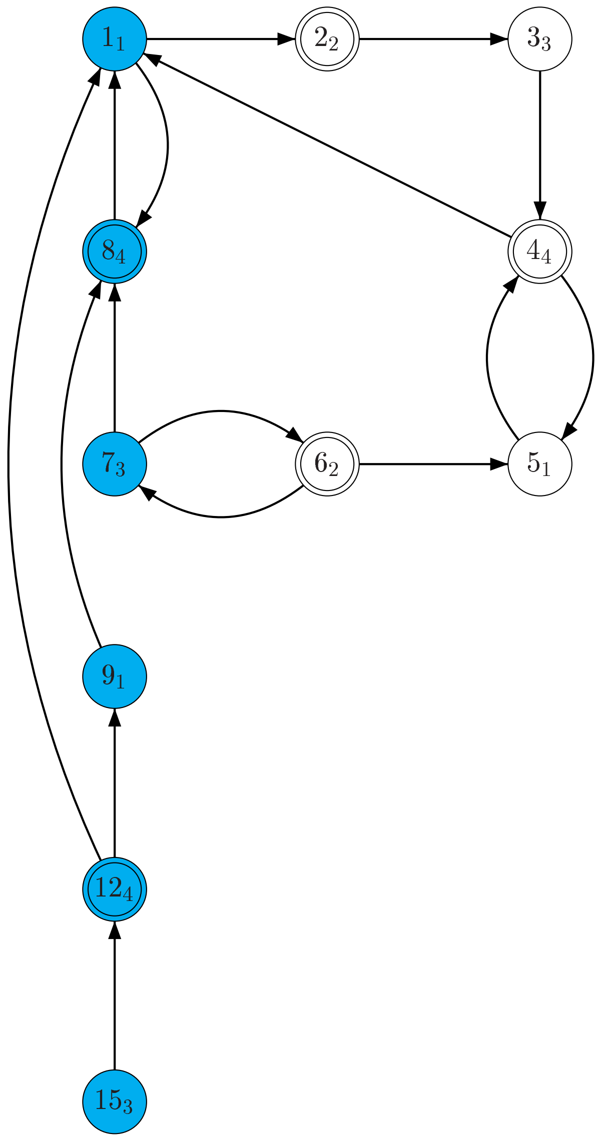
(d) $N_2 = \{v \in Y_2 : \; 4 = \phi(v)\} = \{1_4, 8_4, 12_4\}$



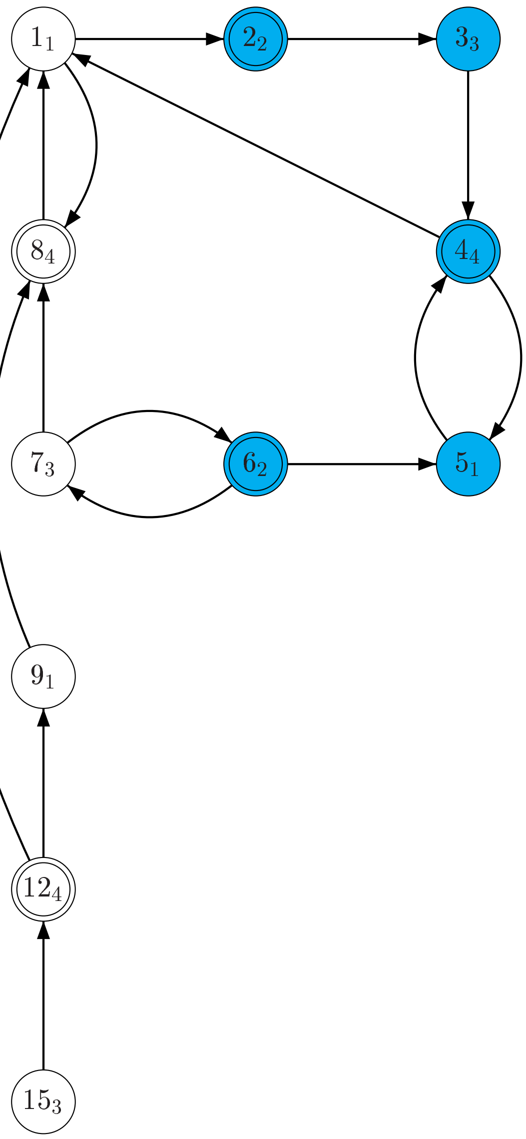
(e) $\mathrm{Attr}^0(G[Y_2], N_2)$



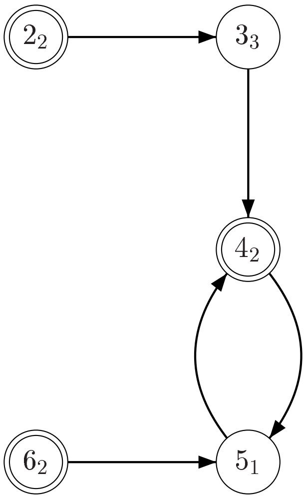
$\mathrm{Attr}^0(G[Y_2], N_2) =$



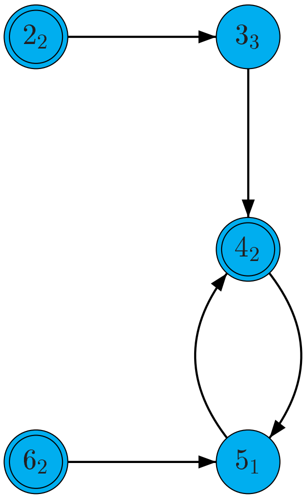
(f) $Z_2 = Y_2 \setminus \mathrm{Attr}^0(G[Y_2], N_2) =$



(g) We concentrate on the game played on $G[Z_2]$:

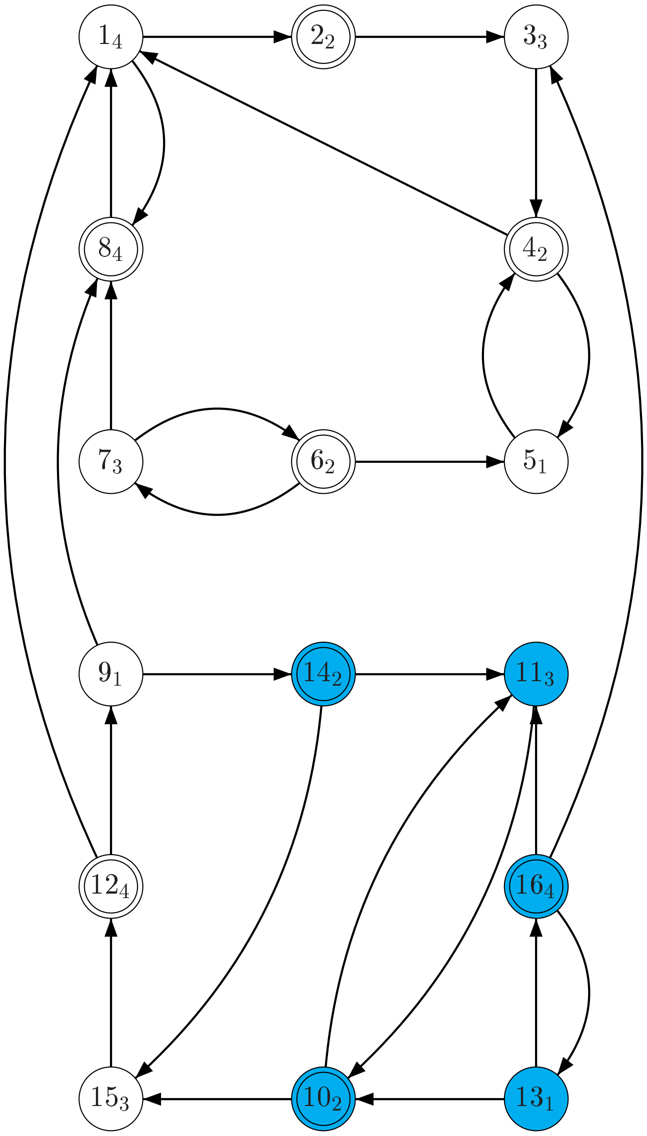


(h) Once more, one easily gets Z_2^0 the set of winning vertices for 0, and the coresponding winning memoryless strategy. Of course, $Z_2^1 = \emptyset$

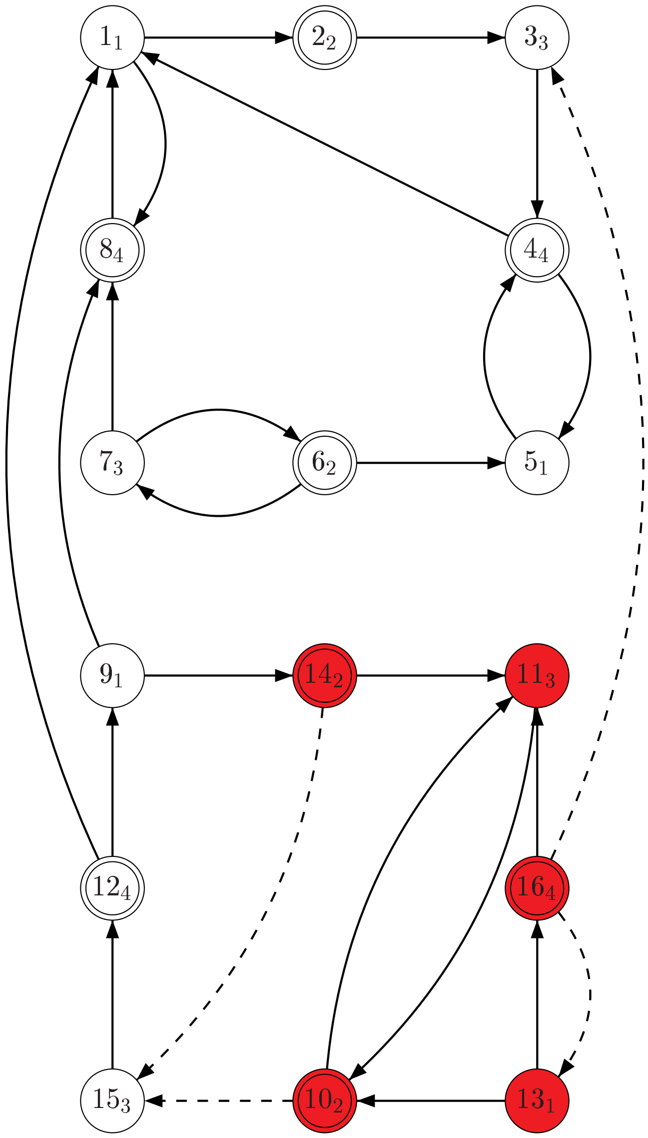


(4) (a) $W_3^1 = X_2 \cup Z_2^1 = \{10_2, 11_3, 13_1, 14_2, 16_4\} \cup \emptyset = \{10_2, 11_3, 13_1, 14_2, 16_4\} = W_2^1.$

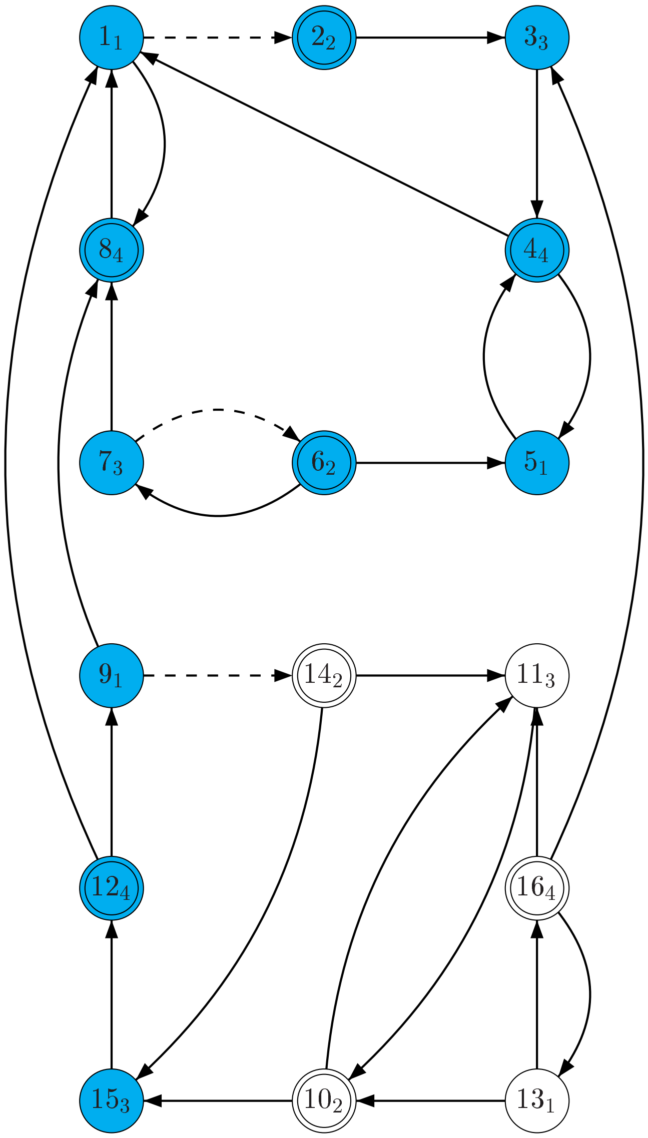
$= W^1$ the winning set for player 1.



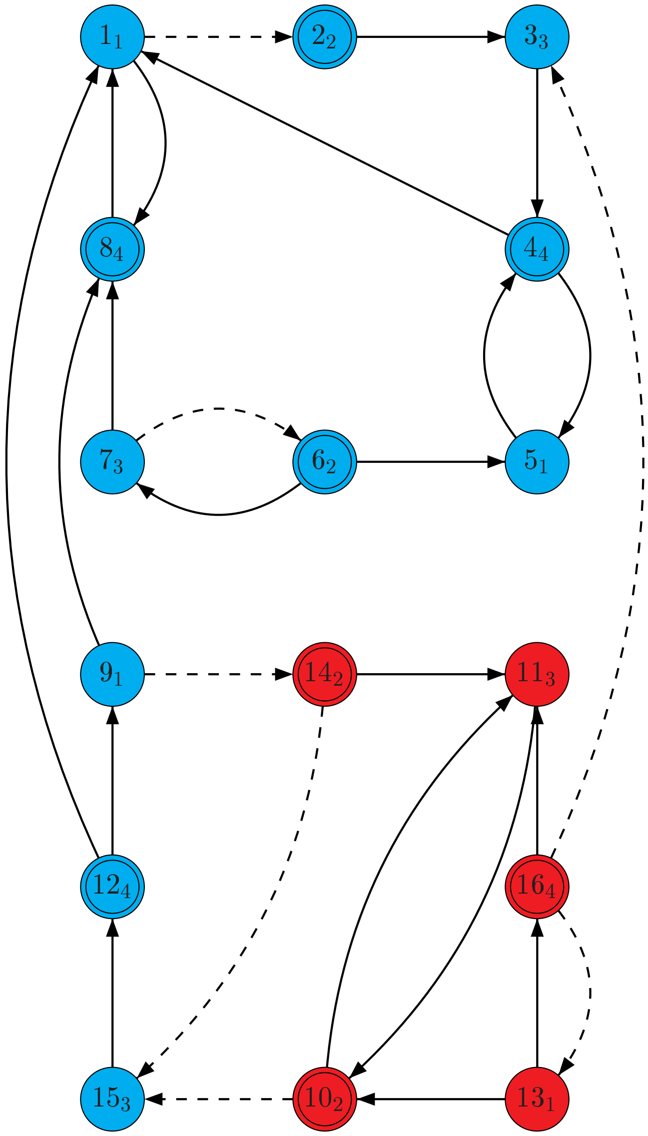
(1) A memoryless winning strategy for 1 on his winning set W^1:



(2) A memoryless winning strategy for 0 on his winning set W^0:



(3) Both memoryless winning strategies for 0 and 1 on their respectives winning sets W^0 and W^1:



The algorithm derived from the proof of theorem 56 computes the set W^0 of starting positions winning for player 0 in exponential time. The question is: how complicated is the problem, given a parity game G and a vertex v, to determine which player has a winning strategy when starting the play from this vertex?

Notation 59

PARITY-GAMES $= \{(G, v) : G$ *a parity game,* $v \in V_0 \cup V_1$*, player* **0** *has a w.s. from* $v\}$.

Proposition 60

$$\textbf{PARITY-GAMES} \in \textbf{NP} \cap \textbf{CO-NP}.$$

Proof of proposition 60: Remark first that, given G and $p \in \mathbb{N}$, the problem to determine whether there exists or not a loop in G with maximal priority p can be computed in polynomial time. We recall that **REACHABILITY** $\in$ **PTIME**. Given G' any subgraph of (V_0, V_1, E), we write $v \underset{G'}{\longmapsto} w$ to say that there is a path from v to w in G'. We also write $G_{\leq p}$ to denote the subgraph induced by $\phi^{-1}\{0, \dots, p\}$.
There is a loop with maximal priority p precisely when there exists some $v_p \in \phi^{-1}\{p\}$ such that

$$\{v \in \phi^{-1}\{0, \dots, p\} : v \underset{G_{\leq p}}{\longmapsto} v_p\} \bigcap \{v \in \phi^{-1}\{0, \dots, p\} : v_p \underset{G_{\leq p}}{\longmapsto} v\} \neq \emptyset.$$

One can define the following nondeterministic algorithm to determine whether player 0 has a w.s. from v inside G:

Inputs : G, v

Guess σ a memoryless strategy for 0 (i.e. for each $v' \in V_0$, pick one $v' \in vE$)

$G_\sigma :=$ the subgraph induced by σ (i.e. edges not chosen by σ are removed)

for each v' s.t. $\phi(v')$ is odd **do**

for each $v \in G_{\sigma \leq \phi(v')}$ **do**

if $\left(v \underset{G_{\sigma \leq \phi(v')}}{\longmapsto} v' \quad \textbf{and} \quad v' \underset{G_{\sigma \leq \phi(v')}}{\longmapsto} v \quad \textbf{and} \quad v \underset{G_\sigma}{\longmapsto} v'\right)$

reject

accept .

This algorithm clearly works in polynomial time in the size of the inputs. This shows the problem belongs to **NP**. To show that it is in **CO-NP**, it is enough to prove that

$$\{(G, v) : G \text{ a parity game, } v \in V_0 \cup V_1, \text{ player } \mathbf{1} \text{ has a w.s. from } v\} \in \textbf{NP}.$$

The same algorithm as above, except that player 0 is turned into 1, and odd is replaced by even gives the result.

$\dashv$ 60

There is even a better result: **PARITY-GAMES** $\in$ **UP** $\cap$ **CO-UP**. Where **UP** is the class of all problems solvable by an *unambiguous* nondeterministic Turing machine working in polynomial time. (A nondeterministic Turing machine is *unambiguous* if for any input there is at most one accepting computation).

Parity games are of major interest in computer science for they are directly related to the model-checking of μ-calculus (L_μ). This logic is obtained by adding least and greater fixpoints to propositional modal logic (ML). μ-calculus, just like Modal Logic is evaluated on transition systems (Kripke models = colored graphs). The model checking problem for μ-calculus is: given such a transition system $\mathcal{G}$, a node v, and a formula ψ, to decide whether $\mathcal{G}, v \models \psi$ (ψ holds at node v) or not. Basically, given such a triple $(\mathcal{G}, v, \psi)$ one can build in polynomial time (in the size of $(\mathcal{G}, v, \psi)$), a parity game G such that for a certain given vertex w:

$$\mathcal{G}, v \models \psi \iff \text{player 0 has a w.s. in the parity game } G \text{ from vertex } w.$$

The idea of the construction of the arena G:

- vertices represent subformulas of ϕ, in such a way that if there is no least fixpoint nor greatest fixpoint operator in ϕ, (the game looks like an Hintikka game).
- Loops in G stand for the existence of least or greatest fixpoint operators in ϕ.
- The number of priorities (colors) necessary to define G witnesses the alternation depth of least and greatest fixpoint operators inside ϕ.

Remark 61

$$\textbf{PARITY-GAMES} \in \mathbf{P} \implies \text{Model-Checking } (L_\mu) \in \mathbf{P}$$

$$\textbf{PARITY-GAMES} \notin \mathbf{P} \implies \mathbf{P} \neq \mathbf{NP}.$$

Unfortunately the question "PARITY-GAMES ∈ P ?" remains open. Solving this problem in either way would be an important breakthrough in Theoretical Computer Science.

We come back to games played on an arena with general Muller conditions. The question being: how much memory is neede to win a game. More precisely, the winning player has a winning strategy and in fact always a winning strategy that relies on two things: the vertex he visits and some bounded memory that depends on what particular vertices where visited before. The precise answer to that question is the entire subject of next section.

4.2.2 Later Appearance Record

Definition 62 *Given C a set of colors and $B \subseteq C$,*
the Later Appearance Record for B is a finite deterministic transition system

$$LAR_B = (C, M_B, \delta_B)$$

where

(1) C is the alphabet,

(2) $M_B = \{u \in B^{|B|} :$ each color $c \in B$ appears exactly once in $u\}$ is the set of states:

(3) $\delta_B : M_B \times C \longmapsto M_B$ *is the transition function defined by:*

$$\delta_B(u,c) = \begin{cases} u_1 u_2 c & if \quad c \in B \text{ and } u = u_1 c u_2 \text{ for some } u_1, u_2 \in B^* \\ u & if \quad c \notin B \end{cases}$$

To describe conveniently how a player's memory is updated, we assume that the coloring mapping ϕ is extended to a total coloring on $C \cup \{\varepsilon\}$ where ε stands for the "color" of uncolored nodes. We also set $\delta_B(u,\varepsilon) = u$, for any $u \in M_B$
Suppose that the memory used by player i is a LAR$_B$ memory M_B for some $B \subseteq C$, his current memory state is $u \in M_B$, and that the token stands in a vertex v. If the token is moved to a successor vertex $w \in vE$, then player i updates his memory to the new state $\delta_B(u,\phi(w))$. In particular, the memory remains the same if w is uncolored or colored by some $c \in C \setminus B$.

Definition 63 *Given an arena* $G = (V_0, V_1, E, C, \phi)$ *a set of colors and* $B \subseteq C$,

(1) A LAR$_B$ *strategy for player* i *is a mapping:*

$$\sigma^i : V_i \times M_B \longmapsto V_{1-i}.$$

(2) A LAR$_B$ *history is a sequence*

$$h = (v_0,u_0),(v_1,u_1),(v_2,u_2),\ldots,(v_i,u_i),(v_{i+1},u_{i+1}),\ldots\ldots$$

of consecutive token positions $v_i \in V$ *and LAR*$_B$ *memory states* $u_i \in M_B$ *such that:*

(a) $v_0 v_1 v_2 \ldots v_i v_{i+1} \ldots\ldots$ *is a play in* G, *and*

(b) $\forall i \in \mathbb{N} \quad u_{i+1} = \delta_B(u_i, \phi(v_{i+1}))$.

Notation 64 *Given* $\mathcal{F} \subseteq (\mathcal{P}(C) \setminus \{\emptyset\})$,

$$USF(\mathcal{F}) = \bigcup_{\substack{X_1, X_2 \notin \mathcal{F} \\ X_1 \cup X_2 \in \mathcal{F}}} X_1 \Delta X_2$$

where $X_1 \Delta X_2 = (X_1 \setminus X_2) \cup (X_2 \setminus X_1)$.

Theorem 65 *Given an arena* $G = (V_0, V_1, E, C, \phi)$ *and* $(\mathcal{F}_0, \mathcal{F}_1)$ *Muller conditions for games played on* G,

(1) W^i *is a* $(1-i)$*-trap in* G,

(2) player i *has a winning LAR*$_{USF(\mathcal{F}_i)}$ *strategy on* W^i.

Proof of theorem 65: Very similar to theorem 56, but much more complicated (due to the presence of the $USF(\mathcal{F}_i)$ memory.

$\dashv$ 65

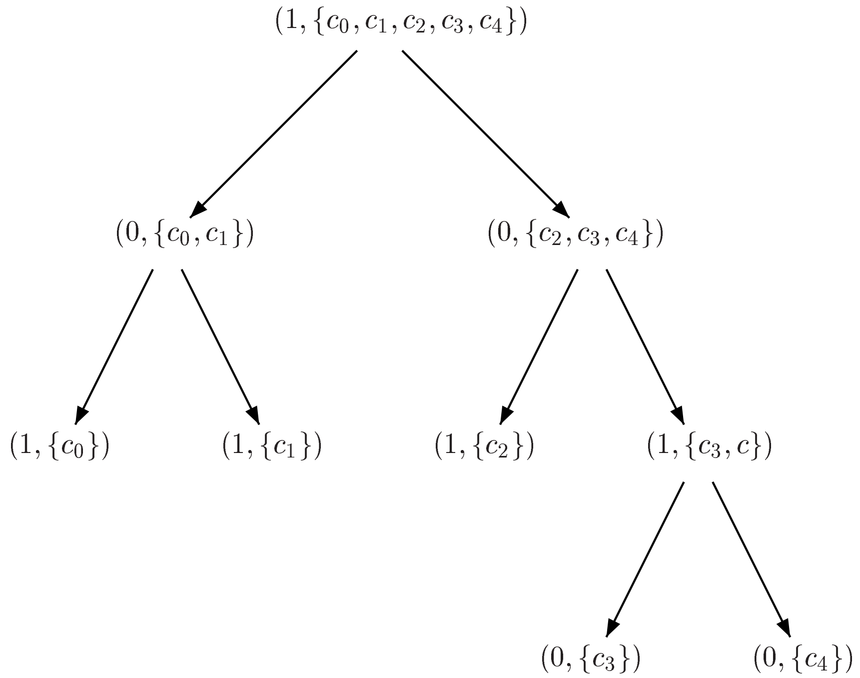
Definition 66 *Given an arena* $G = (V_0, V_1, E, C, \phi)$ *and* $(\mathcal{F}_0, \mathcal{F}_1)$ *Muller conditions for games played on* G,
the **split tree** *associated with* $(\mathcal{F}_0, \mathcal{F}_1)$ *is a finite tree* $\mathcal{T}_{(\mathcal{F}_0, \mathcal{F}_1)}$ *whose nodes are couples* (i, S) *with* $i \in \{0,1\}$ *and* $S \in \mathcal{F}_i$.

- *The root of* $\mathcal{T}_{(\mathcal{F}_0,\mathcal{F}_1)}$ *is* (i, C) *where* i *is the integer such that* $C \in \mathcal{F}_i$
- *if* (j, S) *is a node in* $\mathcal{T}_{(\mathcal{F}_0,\mathcal{F}_1)}$ *and* $S_0, \ldots, S_k$ *are the maximal (for inclusion) subsets of* S *belonging to* $\mathcal{F}_{1-j}$*, then* (j, S) *has* k *sons:*

$$(1-j, S_0), \ldots, (1-j, S_k).$$

- *Otherwise* (j, S) *is a leaf.*

Example 67 Let $C = \{c_0, c_1, c_2, c_3, c_4\}$, $\mathcal{F}_0 = \{\{c_0, c_1\}, \{c_2, c_3, c_4\}, \{c_2, c_3\}, \{c_2, c_4\}, \{c_3\}, \{c_4\}\}$, $\mathcal{F}_1 = (\mathcal{P}(C) \smallsetminus \{\emptyset\}) \smallsetminus \mathcal{F}_0$,
the split tree for $(\mathcal{F}_0, \mathcal{F}_1)$ is :



Remark 68 Given an arena $G = (V_0, V_1, E, C, \phi)$ and $(\mathcal{F}_0, \mathcal{F}_1)$ Muller conditions for games played on G,

$$\Big(\emptyset \neq Y \subseteq C \ \wedge \ Y \in \mathcal{F}_i\Big)$$

$$\Longleftrightarrow$$

$$\text{there exists } (i, S) \in \mathcal{T}_{(\mathcal{F}_0,\mathcal{F}_1)} \ \Big(Y \subseteq B \ \bigwedge \ \Big((1-i, S') \text{ is a child of } (i, S) \longrightarrow Y \not\subseteq S'\Big)\Big)$$

Proof of remark 68:

$(\Longrightarrow)$: straightforward by induction on the height of the split tree.

$(\Longleftarrow)$: is obvious.

$\dashv$ 68

Lemma 69 *Given an arena $G = (V_0, V_1, E, C, \phi)$ and $(\mathcal{F}_0, \mathcal{F}_1)$ Muller conditions for games played on G,*

$$\mathcal{F}_{1-i} \textit{ is closed under union } \iff \textit{ each } i\textit{-vertex of } \mathcal{T}_{(\mathcal{F}_0,\mathcal{F}_1)} \textit{ has at most one child}$$

Proof of lemma 69:

$(\Longrightarrow)$: two different children of a i-vertex would constitute a split in $\mathcal{F}_{1-i}$.

$(\Longleftarrow)$: Assume that all i-nodes have at most one child. Suppose that $\mathcal{F}_{1-i}$ contains a split $X, Y \in \mathcal{F}_{1-i}$ and $X \cup Y \in \mathcal{F}_i$. Then, by remark 68, there exists a node (i, S') in $\mathcal{T}_{(\mathcal{F}_0,\mathcal{F}_1)}$ with only child $(1-i, S'')$ such that $(X \cup Y \subseteq S')$ and $(X \cup Y \not\subseteq S'')$. However, S'' is the greatest subset of S' belonging to $\mathcal{F}_{1-i}$, in particular X and Y is should be subset of S''. This would lead to $X \cup Y \subseteq S''$, a contradiction.

⊣ 69

Theorem 70 *Let $(\mathcal{F}_0, \mathcal{F}_1)$ be Muller conditions, the following are equivalent:*

(1) player i has a memoryless w.s. (on W^i) on all games with Muller conditions $(\mathcal{F}_0, \mathcal{F}_1)$

(2) $\mathcal{F}_{1-i}$ is closed under union

(3) each i-vertex of $\mathcal{T}_{(\mathcal{F}_0,\mathcal{F}_1)}$ has at most one child.

Proof of theorem 70: (2) $\iff$ (3) is lemma 69. (1)$\Longrightarrow$(2) is proposition 43. To show (2)$\Longrightarrow$(1) it is enough to notice that in case $\mathcal{F}_{1-i}$ is closed under union, then $USF(\mathcal{F}_{1-i})$ is the empty set. Hence, $\text{LAR}_{USF(\mathcal{F}_{1-i})}$ is also reduced to the empty word. Therefore, the memory needed to win the game being empty, player i has a memoryless winning strategy. ⊣ 70

In particular, this proves theorem 45.