Games over finite graphs

Exercise 1

For the whole exercise we assume that we are given a finite graph \( G = (V, E) \), with \( V \) a finite set of vertices and \( E \subseteq V \times V \) a finite subset of edges. Moreover, we assume that there is no dead-end (that is for all \( v_1 \in V \) there exists \( v_2 \in V \) such that \((v_1, v_2) \in E\)). We additionally fix a partition \( V_E \cup V_A \) of the vertices \( V \) (that is \( V_E \cap V_A = \emptyset \) and \( V_E \cup V_A = V \)). This partition is then used to define an arena \( G = (G, V_E, V_A) \) as explained during the lectures.

Question 1: Let \( F_1, \ldots, F_n \) be a collection of subsets of \( V \) that are pairwise disjoint (for all \( i \neq j, F_i \cap F_j = \emptyset \)). We consider the following winning condition for Eve:

\[
\Omega = V^* F_1 V^* F_2 \cdots V^* F_n V^\omega
\]

and we denote by \( G = (G, \Omega) \) the associated game. Hence in this game, a play \( \lambda = v_0 v_1 v_2 \cdots \) is won by Eve if and only if the sets \( F_1, \ldots, F_n \) are visited, in this order, at least one, i.e., there are \( i_1 < i_2 < \cdots i_n \) such that for all \( 1 \leq j \leq n \), \( v_{i_j} \in F_j \).

(a) Give an algorithm that decide for a given vertex whether it is winning for Eve.

(b) What is the complexity of your algorithm?

(c) Does Eve always has a winning positional strategy from any winning vertex in \( G \)?

(d) Can you build a winning strategy for Eve from any winning vertex for her in \( G \)?

Answer. We define the following sets: \( W_{n+1} = V \) and \( W_i = \text{Attr}(W_{i+1} \cap F_i) \) for all \( 1 \leq i \leq n \), where \( \text{Attr} \) is the usual attractor for Eve as seen in the course. We have the following property:

For all \( 1 \leq i \leq n + 1 \), \( W_i \) consists of those vertices from which Eve has a winning strategy in the game whose winning condition is \( \Omega_i = V^* F_i V^* F_{i+2} \cdots V^* F_n V^\omega \) (with the convention that \( \Omega_{n+1} = V^\omega \)). Then the property is proved by (decreasing) induction on \( i \). For \( i = n + 1 \) the property is obvious. Then assume the property holds for some \( i + 1 \). By definition of the attractor, Eve has from \( W_i = \text{Attr}(W_{i+1} \cap F_i) \) a positional strategy \( \varphi \) to reach the set \( W_{i+1} \cap F_i \) and by induction hypothesis, she has a strategy \( \varphi \) to ensure, from any vertex in \( W_{i+1} \), that the play belongs to \( \Omega_{i+1} \). Now consider the strategy \( \varphi_{i+1} \) of Eve that is defined as follows: play accordingly to \( \varphi \) as long as \( W_{i+1} \cap F_i \) is not reached and then play accordingly to \( \varphi_{i+1} \) forever. Consider a play \( \lambda \) that starts in \( W_i \) and where Eve respects \( \varphi_{i+1} \): first, it eventually visits \( W_{i+1} \cap F_i \) (as Eve first respects \( \varphi \)), and is of the form \( \lambda_1 v \lambda_2 \) with \( v \in W_{i+1} \cap F_i \) and \( \lambda_1 \) not visiting \( W_{i+1} \cap F_i \). Then \( \nu \lambda_2 \) is a play where Eve respects \( \varphi_{i+1} \), hence \( \lambda_2 \in \Omega_{i+1} \), hence \( \lambda \in \Omega_i \). This proves that from any vertex in \( W_i \) Eve has a strategy to ensure that the resulting play is in \( \Omega_i \). Conversely, consider now a vertex \( v \notin W_i \). By definition, Adam has a strategy \( \psi \) to prevent reaching \( W_{i+1} \cap F_i \), and (by induction hypothesis), for any element not in \( W_{i+1} \) Adam has a strategy \( \psi \) to prevent to produce a play in \( \Omega_{i+1} \). Now consider the following strategy for Adam: play according to \( \psi \) as long as \( F_i \) is not visited, and if this eventually happens, then play according to \( \psi_i \). Consider a play \( \lambda \) starting from a vertex not in \( W_i \) and where Adam respects \( \psi \) : then either \( \lambda \) never reach \( F_i \) (hence \( \lambda \notin \Omega_i \)) or \( \lambda = \lambda_1 v \lambda_2 \) with \( v \in F_i \) (hence \( v \notin W_{i+1} \)) and \( \lambda_1 \) not visiting \( F_i \) and \( v \lambda_2 \) is a play where Adam respects \( \psi_i \). Hence \( \lambda \notin \Omega_i \) and therefore \( \lambda \notin \Omega_i \). This concludes the induction. Thus the winning vertices in the original game are those in \( W_1 \).
Computing $W_1$ requires $n$ computation of the attractor set, which can be achieved in $O(n \times |E|)$.

Eve may not have a positional strategy in the game. For instance consider a three vertex game where Eve is playing alone and such that the vertices are $u,v$ and $w$ and the edges $\{(u,v)(v,u)(u,w)(w,u)\}$ and $F_1 = \{v\}$ and $F_2 = \{w\}$. Then any positional strategy will produce a play that visits only $\{(u,v)\}$ or only $\{(u,w)\}$, hence loosing. Of course Eve has winning strategy in this game.

From the proof above, one can construct a winning strategy (as $\varphi_i$ is built from $\varphi_{i+1}$ and $\varphi$ a positional strategy in a reachability game). In particular one can note that the resulting strategy only uses finite memory (of size $n$).

**Question 2:** We again consider a collection $F_1, \cdots, F_n$ of subsets of $V$ that are pairwise disjoint (for all $i \neq j$, $F_i \cap F_j = \emptyset$). We consider the following winning condition for Eve:

$$\Omega = \bigcap_{i=1}^{n} V^* F_i V^\omega$$

and we denote by $G = (G, \Omega)$ the associated game. Hence in this game, a play $\lambda = v_0 v_1 v_2 \cdots$ is won by Eve if and only if the sets $F_1, \cdots, F_n$ are visited, in any order, at least one, i.e. there are $i_1, i_2, \cdots, i_n$ such that for all $1 \leq j \leq n$, $v_{i_j} \in F_{i_j}$.

(a) Give an algorithm that decide for a given vertex whether it is winning for Eve. One might design a new equivalent game (with a simpler winning condition studied in the course).

(b) What is the complexity of your algorithm?

(c) Does Eve always has a winning positional strategy from any winning vertex in $G$?

(d) Can you build a winning strategy for Eve from any winning vertex for her in $G$?

**Answer.** We define a new graph $G' = (V', E')$ where $V' = V \times 2^{\{1, \cdots, n\}}$ and $E'$ consists of the pairs $((v_1, S_1), (v_2, S_2))$ such that $(v_1, v_2) \in V$ and $S_2 = S_1 \cup \{i \mid v_2 \in F_i\}$. We partition $V'$ by letting $V'_E = V_E \times 2^{\{1, \cdots, n\}}$, which leads to define an arena $G'$. Then one define a set of final vertices $F' = V \times \{1, \cdots, n\}$ and let $G'$ be the reachability game induced by $F'$ on $G'$. We claim that Eve has a winning strategy in $G$ from a vertex $v$ if she has a winning strategy in $G'$ from $(v, \{i \mid v \in F_i\})$. One can note that in $G'$, from some vertex $(v, S)$ if there is an edge to some $(v', S')$ then $S'$ is uniquely determined from $v'$ and $S$. In particular, it means that any strategy $\varphi$ in $G$ can be lifted to a strategy in $G'$ by letting $\varphi'((v_1, S_1) \cdots (v_k, S_k)) = (v, S)$ where $v = \varphi(v_1 \cdots v_k)$ and $S$ is the unique possible set. Assume that Eve wins in $G$ from $v$ and call $\varphi$ a winning strategy. Consider a (finite) play $\lambda = (v_1, S_1)(v_2, S_2) \cdots$ in $G'$ from $(v, \{i \mid v \in F_i\})$ where $S_2 = S_1 \cup \{i \mid v_2 \in F_i\}$ (this is independent of $\varphi'$). Now by definition of $\varphi'$, one has that $v_{i_1} v_{i_2} \cdots$ is a play in $G$ where Eve respects $\varphi$, hence it is winning, hence there is some $k$ such that $S_k = \{1, \cdots, n\}$, meaning that $\varphi'$ is winning.

Now assume that Eve wins in $G'$ from $(v, \{i \mid v \in F_i\})$ and call $\varphi'$ a winning strategy. One can assume that $\varphi'$ is positional as $G'$ is a reachability game. Define a strategy $\varphi$ in $G$ by letting $\varphi(v_0, \cdots v_k) = v_{k+1}$ where $(v_{k+1}, S_{k+1}) = \varphi'(v_k, \{i \mid \exists j \leq k \text{ s.t. } v_k \in F_i\})$. Consider a play $v_0 v_1 \cdots$ where Eve respects $\varphi$. Then the play $(v_0, S_0)(v_1, S_1) \cdots$ where we let $S_k = \{i \mid \exists j \leq k$, $v_k \in F_i\}$...
ks.t. \(v_k \in F_1\) is a play in \(\mathcal{G}\) where Eve respects \(\varphi'\). Therefore it is a winning play, meaning that \(v_0v_1\cdots\) is winning for Eve in \(\mathcal{G}\). This concludes the proof.

Complexity is exponential as the game \(\mathcal{G}'\) is (and solving a reachability game is linear).

There is no memoryless strategy in general for the same reason as previously. From the proof it follows that one can always construct a winning strategy in \(\mathcal{G}\) (as one can built one in \(\mathcal{G}'\) and lift it back to \(\mathcal{G}\).

**Question 3:** We again consider a collection \(F_1, \ldots, F_n\) of subsets of \(V\) that are pairwise disjoint (for all \(i \neq j, F_i \cap F_j = \emptyset\)). We consider the following winning condition for Eve: a play \(\lambda = v_0v_1v_2\cdots\) is won by Eve if and only if the each of the set \(F_1, \ldots, F_n\) is visited infinitely often. Equivalently, for all \(1 \leq j \leq n\) there exists \(i_1 < i_2 < i_3 \cdots\) such that for all \(1 \leq k\), \(v_{i_k} \in F_j\).

(a) Give an algorithm that decide for a given vertex whether it is winning for Eve. One might design a new *equivalent* game (with a simpler winning condition studied in the course).

(b) What is the complexity of your algorithm?

**Answer.** It suffices to remark that the winning condition is unchanged is one force the order in which the sets \(F_i\) are visited, \(i.e.\) she has to infinitely visit \(F_1\) and then \(F_2\) and then \(F_3 \cdots F_n\). Hence it suffices to build an new game with an extra component recalling which \(F_i\) should be visited next, and to define as a winning condition to infinitely often switch from the component being \(n\) to the component being \(1\).

**Question 4:** [Difficult] We go back to the setting of the second question, but we additionally assume that all the \(F_j\) are singleton. Give a *polynomial time* algorithm to decide whether a vertex is winning for Eve.

**Answer.** For all pair \((i, j)\) of vertices one check whether \(i\) belongs to the attractor of \(j\). Then one builds a graph whose vertices are \(1, \ldots, n\) and where there is an edge from \(i\) to \(j\) if and only if \(i\) belongs to the attractor of \(j\). Then in this graph there is a path that visits all vertices if and only if Eve has a winning strategy (this is easy to check). Looking for such a path is checked in polynomial time (one essentially builds the strongly connected components of the graph).