April 22, 2022

Mean payoff games with the signature of a potential

Marianne Akian, Stéphane Gaubert, Oliver Lorscheid, and Matthias Mnich

Abstract. In this paper, we develop an algorithm that finds the winning positions for a mean payoff game in terms of a reduction to mean payoff games on subgraphs by means of a reachability game. For certain subclasses of games, this produces a significantly faster algorithm compared to previous methods. In particular, mean payoff games on bipartite graphs that have the "signature of a potential" can be solved in linear time. This includes all parity games on a symmetric and bipartite graph.

Contents

Introduction

Mean payoff games are two player deterministic zero-sum games with full information that were introduced by Ehrenfeucht and Mycielski in [2]. They form an interesting problem in complexity theory since they are known to be in NP and co-NP, but until now they withstand a solution in polynomial time. Since their definition by Ehrenfeucht and Mycielski, several variations of mean payoff games have been considered, which are not essential from the viewpoint of complexity theory since they can all be deduced from each other in polynomial time.

In this text, we develop an algorithm in the following setting of a mean payoff game, which is given as follows: two players Min and Max play on a finite and directed graph G with vertex set V and edge set E together with a partition $V = V_+ \sqcup V_-$ and a weight function $\mu : E \to \mathbb{Z}$. A move at a vertex v in V is the choice of an edge $(v,w) \in E$ where Min decides the move if $v \in V_-$, and otherwise Max decides. A play is a finite or infinite sequence of moves of the form $(v_1, v_2), (v_2, v_3), \dots$, also written as the sequence of vertices $(v_1, v_2, v_3, \dots)$ where we assume that $(v_1, \dots, v_n)$ is finite if v_n is without outgoing edges.

In the case that $(v_1, \dots, v_n)$ is finite, Min wins if $v_n \in V_+$ and Max wins if $v_n \in V_-$. If $\pi = (v_1, \dots)$ is infinite, then the sequence of mean values

$$\chi_n(\pi) \;=\; \tfrac{1}{n} \cdot \sum_{i=1}^{n} \mu(v_i, v_{i+1})$$

is limited from below and above by the smallest and largest value of $\mu : E \to \mathbb{Z}$, respectively. In particular, this series has an inferior and superior limit

$$\chi_-(\pi) \;=\; \liminf_{n\to\infty} \chi_n(\pi) \qquad \text{and} \qquad \chi_+(\pi) \;=\; \limsup_{n\to\infty} \chi_n(\pi),$$

respectively. In the infinite case, Min wins if $\chi_+(\pi) < 0$, Max wins if $\chi_-(\pi) > 0$, and otherwise the play is a draw.

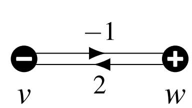
Ehrenfeucht and Mycielski show in [2] that mean payoff games allow for optimal positional strategies for Min and Max and that every vertex $v \in V$ has a characteristic value $\chi(v)$, which satisfies that $\chi_-(\pi) = \chi_+(\pi) = \chi(v)$ for every play $(v_1, \dots)$ starting in $v_1 = v$ and following optimal strategies for both players.

It has been the subject of several studies to develop algorithms that can decide which player has a winning strategy at a given vertex v, that determine the characteristic value at v and that find optimal strategies for the players; see [4] and others. In this text, we combine such an algorithm with other known methods, which leads to an algorithm that determines all winning positions for each player. This algorithm improves on existing methods and yields the result in linear time for the subclass of mean payoff games with the signature of a potential (to be defined below).

The central idea. Consider a mean payoff game with vertex set $V = V_+ \sqcup V_-$, edge set E and weight function $\mu : E \to \mathbb{Z}$. The algorithm that we present in this paper finds the set $\mathbf{W}_+$ of vertices v for which Max wins (for plays starting in v following an optimal strategy), the set $\mathbf{W}_-$ of vertices v for which Min wins and the set $\mathbf{W}_0$ of vertices v for which both of Min and Max can achieve a draw. We define $\gamma(v)$ as the element of $\{+,-,0\}$ such that $v \in \mathbf{W}_{\gamma(v)}$. We also say that *Max wins at v* if $\gamma(v) = +$ and that *Min wins at v* if $\gamma(v) = -$.

The central idea of our algorithm is based on the following insights:

(1) If (v,w) is an edge of G such that, say, player Min plays at v and such that Max can force the play to return to v with positive payoff for the whole cycle from v to v, then it is not lucrative for Min to play along (v,w). This means that (v,w) is *obsolete* for determining the value of $\gamma(v)$, which reduce the complexity of the mean payoff game by omitting the edge (v,w); cf. Figure 1.



mean payoff: $\frac{1}{2}(-1+2) = \frac{1}{2} > 0$

Figure 1. Obsolete edge (v,w) for Min

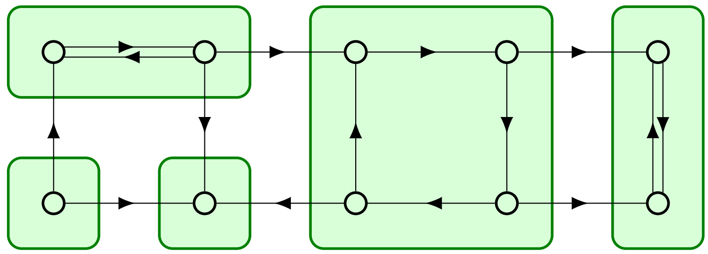


Figure 2. The strongly connected components of a graph

(2) Let X be the vertex set of a subgraph of V that is *successor closed*, i.e. for every edge $(v,w) \in E$ with $v \in X$ also $w \in X$. Then the winning positions of the mean payoff game in X (w.r.t. the restrictions of V_+, V_-, E and μ to X) are also winning positions of V. Simplifying matters, this allows us to deduce the winning positions for V from the winning positions of its *strongly connected components*, which are the maximal subgraphs of G that contain a directed path from each of its vertices to any other of its vertices; cf. Figure 2.

(3) If we know that one of the players, say Max, wins at a vertex v and if Max can force every play starting in another vertex w to pass through v, then Max also wins at w. Thus if we know some winning positions, we can find more winning positions by solving a *reachability game* on the graph G; cf. Figure 3.

Remark on illustrations. In pictures, we mark vertices in V_+ with a "$+$" and vertices in V_- with a "$-$". We draw an empty circle for vertices of V for which we do not specify to which player they belong.

A complete reduction to reachability games. As a warm up, we consider the following particular class of mean payoff games for which the deletion of obsolete edges yields a complete reduction to a reachability game.

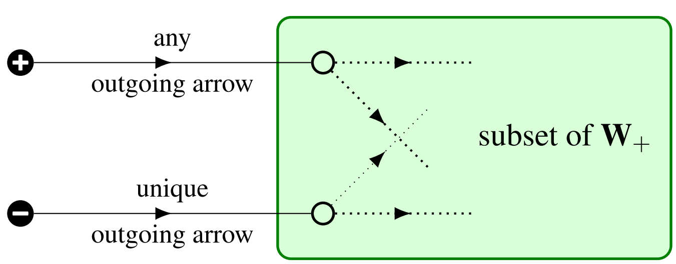


Figure 3. Finding additional winning vertices of Max via reachability

Definition. A *potential on G* is a function $\psi : V \to \mathbb{Z}$. The mean payoff game on G has *the signature of a potential $\psi : V \to \mathbb{Z}$* if for all edges $(v,w) \in E$,

(P0) $$\mu(v,w) \;=\; \mu(w,v) \;=\; \psi(v) - \psi(w) \;=\; 0 \qquad \text{if } \epsilon(v) = \epsilon(w),$$

(P1) $$\operatorname{sign}\big(\epsilon(v)\mu(v,w) - \epsilon(w)\mu(w,v)\big) \;=\; \operatorname{sign}\big(\psi(v) - \psi(w)\big) \qquad \text{if } \epsilon(v) \neq \epsilon(w).$$

where we define $\operatorname{sign}\big(\epsilon(v)\mu(v,w) - \epsilon(w)\mu(w,v)\big) = -1$ if there is no edge from w to v.

Remark. Axiom (P0) is very restrictive for edges between vertices v and w that belong to the same player $P_{\epsilon(v)} = P_{\epsilon(w)}$. This might be seen, however, not as the main restriction since every mean payoff games can be reduced to a mean payoff game on a bipartite graph in polynomial time, and for bipartite graphs axiom (P0) is vacuous.

Definition. The *derived reachability game of G* is the reachability game on the graph G' with vertex set $V' = V$ and edge set

$$E' \;=\; \big\{(v,w) \in E \,\big|\, \epsilon(v) = \epsilon(w) \text{ or } \epsilon(v)\big(\mu(v,w) + \mu(w,v)\big) > 0\big\},$$

with respect to the decomposition of V' into $V'_+ = V_+$ and $V'_- = V_-$ and the winning sets

$$\mathbf{Z}_+ \;=\; \{v \in V_- \mid v \text{ is a sink in } G'\} \qquad \text{and} \qquad \mathbf{Z}_- \;=\; \{v \in V_+ \mid v \text{ is a sink in } G'\}$$

for Max and Min, respectively.

Theorem A. *Let G be a mean payoff game that has the signature of a potential. Then the winning positions for the mean payoff game G are equal to the winning positions of the derived reachability game of G. Therefore the mean payoff game on G can be solved in linear time in $\#V + \#E$.*

Example. A source for mean payoff games with the signature of a potential are parity games on bipartite symmetric graphs, as can be seen as follows.

Let $\nu : V \to \mathbb{N}_{>0}$ be the function that defines the parity game on G with respect to $V = V_+ \sqcup V_-$. This parity game is equivalent to a mean payoff game in which the payment of arc (v,w) is $\mu(v,w) := (-t)^{\nu(v)}$ where t is a suitably large integer (it suffices to take t equal to the number of nodes). Then, this mean payoff game has the signature of the potential function $\psi(v) = \epsilon(v)(-t)^{\nu(v)}$. Berwanger and Serre's result from [1] is recovered as a special case of Theorem A.

Remark. The existence of a potential can be verified in strongly polynomial time, it reduces to finding a point in the relative interior of an alcoved polyhedron.

The general setup. In order to explain Algorithm 1, we introduce some notational conventions. In the following, G is a finite directed graph with vertex set V and edge set E, which we consider as a subset of $V \times V$, i.e. G has a directed edge from v to w precisely when $(v,w) \in E$. In particular, we allow loops (v,v).

We also call Max *player P_+* and Min *player P_-*. The vertex set V comes with a decomposition $V = V_+ \sqcup V_-$ where player P_e determines the move at $v \in V$ if $v \in V_e$ where $e \in \{+,-\}$. Equivalently, $V = V_+ \sqcup V_-$ is characterized by the function $\epsilon : V \to \{+,-\}$

that send v to the value $\epsilon(v) \in \{+,-\}$ with $v \in V_{\epsilon(v)}$. When $\epsilon(v)$ appears in a formula, we identify $+$ with $+1$ and $-$ with -1.

Let $\mu : E \to \mathbb{Z}$ be a weight function on the edge set of G and $M = \max\{|\mu(v,w)| \mid (v,w) \in E\}$ the *size of* μ, i.e. the smallest positive integer such that $[-M,M]$ contains the image of μ. We extend μ to a function $\mu : V \times V \to \mathbb{Z} \cup \{\pm\infty\}$ by defining $\mu(v,w) = -\epsilon(v)\infty$ if $(v,w) \notin E$. A weight $\mu(v,w) = \pm\infty$ can be interpreted as player $P_{\epsilon(v)}$ losing instantly if he or she plays along (v,w). This extension of the domain of μ to all of $V \times V$ is handy for some formulas, for which we use the conventions that $a + (\pm\infty) = \pm\infty$ for all $a \in \mathbb{Z}$ and that $-\infty < 0 < \infty$. Note further that E is characterized by this extended function as the inverse image of $\mathbb{Z}$, and so is the decomposition $V = V_+ \sqcup V_-$ if G does not contain loops since in this case $\mu(v,v) = -\epsilon(v)\infty$. In the following, we consider G as a vertex labelled and edge weighted graph and consider the respective functions ϵ and μ as understood if we refer to the mean payoff game on G.

For $e \in \{+,-\}$, let $V_e^* = \{v \in V_e \mid v \text{ is not a sink}\}$. A *(positional) strategy* for player P_e is a map $\sigma_e : V_e^* \to V$ such that $(v, \sigma_e(v)) \in E$ for all $v \in V_e^*$. A play $\pi = (v_1, \dots)$ *follows the strategy* σ_e if $v_{i+1} = \sigma_e(v_i)$ for all $i \geqslant 1$ such that $\epsilon(v_i) = e$ whenever v_i is not a sink. This should be interpreted that player P_e moves according to the strategy σ_e. A strategy σ_e is *optimal* if every play $\pi = (v_1, \dots)$ that follows σ_e satisfies $e \cdot \chi_{-e}(\pi) \geqslant e \cdot \chi(v_1)$ where $\chi(v_1)$ is the value of the game defined by Ehrenfeucht and Mycielski in [2]. The same paper shows that optimal strategies exist for both players and every mean payoff games.

A play $\pi = (v_1, \dots)$ is *e-optimal* if it follows an optimal strategy σ_e for player P_e. We say that π is *optimal* if it is $+$-optimal and $-$-optimal. In this case, we have $\chi(v_i) = \chi_-(\pi) = \chi_+(\pi)$ for all $i \geqslant 1$. Note that we extend the results from [2] to games with sinks by defining $\chi(v) = -\epsilon(w)\infty$ if there is a finite optimal play π ending in a vertex w. Note further that the characteristic value $\chi(v)$ of v does not depend on the choice of the optimal play π passing through v.

We say that *player P_e wins in v*, and define $\gamma(v) = e$, if $e \cdot \chi(v) > 0$. If $\chi(v) = 0$, then we say that *v is a draw* and write $\gamma(v) = 0$. For $e \in \{+,-,0\}$, we define the subset

$$\mathbf{W}_e \;=\; \{v \in V \mid \gamma(v) = e\}$$

of V, which we call the *set of winning positions for player P_e* if $e \neq 0$ and the *set of draw positions* if $e = 0$. We also call the triple of the three subsets $\mathbf{W}_+$, $\mathbf{W}_-$ and $\mathbf{W}_0$ of V the *winning positions of the mean payoff game*.

Obsolete edges. In the definition of the derived reachability game, we have omitted certain edges which are not relevant to the winning sets of the mean payoff game. In this definition, we only considered the effect of the opponent player playing back via the inverse edge. In the following, we widen this idea by considering all possible plays of bounded length that return to the origin of an edge.

Definition. Let k be a positive integer. An edge $(v,w) \in E$ is *k-obsolete* if for every choices of moves by player $P_{\epsilon(v)}$, player $P_{-\epsilon(v)}$ can achieve that every play $\pi = (v_0, \dots)$

starting $v_0 = w$ either ends in a sink of $P_{\epsilon(v)}$ or passes through v such that $l = \min\{j \geqslant 1 \mid v_j = v\} \leqslant k-1$ and such that

$$\epsilon(v) \sum_{i=0}^{l-1} \mu(v_i, v_{i+1}) \ < \ -\epsilon(v)\mu(v,w).$$

An edge $(v,w) \in E$ is *obsolete* if it is n-obsolete for $n = \#V$.

Proposition B. *Let k be a positive integer and $(v,w) \in E$ an k-obsolete edge. Omitting (v,w) from E does not change the winning positions $\mathbf{W}_+$, $\mathbf{W}_-$ and $\mathbf{W}_0$. The k-obsolete edges of G can be found in $O(k \cdot n^2)$ steps where $n = \#V$, independently from the size M of μ.*

Remark. It is evident from the definition that a k-obsolete edge (v,w) is l-obsolete for all $l \geqslant k$. For small k, we find the following characterizations of k-obsolete edges:

(1) A 1-obsolete edge is a loop (v,v) with $\epsilon(v)\mu(v,v) < 0$ or an edge (v,w) for which w is a sink of $P_{\epsilon(v)}$.
(2) In the case of a bipartite graph, an edge $(v,w) \in E$ is 2-obsolete if $\epsilon(v)\big(\mu(v,w) + \mu(w,v)\big) < 0$ or if there is an edge (w,w') for which w' is a sink.

The case $k = \#V$ is of particular interest since it takes every cycle of G into consideration. More precisely, if (v,w) is an edge in E such that player $P_{-\epsilon(v)}$ can achieve that every play $\pi = (v_1, \dots)$ starting at $v_1 = w$ passes through $v_l = v$ such that $(v_1, \dots, v_l, v_1)$ forms a cycle, i.e. $v_1, \dots, v_l$ are pairwise distinct, and such that

$$\epsilon(v) \sum_{i=1}^{l-1} \mu(v_i, v_{i+1}) \ < \ -\epsilon\mu(v,w),$$

then (v,w) is (n-)obsolete since the length l of a cycle is limited by $n = \#V$.

For an example of a mean payoff game without obsolete edges, cf. Figure 5.

The algorithm. In the following, we describe Algorithm 1 in words. To begin with, we describe the auxiliary functions. Pseudo-code for `Obsolete` can be found in section 2. All other functions are well-known routines. For the sake of completeness, we include pseudo-code for them in Appendix A.

`Obsolete`(k, V_+, V_-, E, μ) **return** F

This function determines the set F of all k-obsolete edges of G.

`StrConnComp`(X, E) **return** $(s, Y_s, \dots, Y_1)$

This function decomposes the subgraph of G with vertex set $X \subset V$ and edge set $E_X = \{(v,w) \in E \mid v,w \in X\}$ into its strongly connected components $Y_1, \dots, Y_s$ with the property that $i \leqslant j$ whenever there is a path from a vertex in Y_j to Y_i. In particular, Y_1 is successor closed.

Algorithm 1: Reduction of a mean payoff game via reachability games

Input: V_+, V_-, E, μ
Output: W_+, W_-, W_0

$F :=$`Obsolete`$(\#V_+ + \#V_-, V_+, V_-, E, \mu)$
while $F \neq \varnothing$ **do**
  $E := E - F$ `// remove obsolete edges`
  $F :=$`Obsolete`$(\#V_+ + \#V_-, V_+, V_-, E, \mu)$
end
$W_+ = W_- = W_0 := \varnothing$ `// initialize the variables`
if $V = \varnothing$ **then** $r := 0$
else $r := 1$
$X_1 := V$
while $r > 0$ **do**
  $X := X_r - (W_+ \cup W_- \cup W_0)$ `// decompose` $X_r - W$
  $(s, X_r, \ldots, X_{r+s-1}) :=$`StrConnComp`$(X, E)$ `// into components`
  $r := r + s - 1$
  $(Y_+, Y_-, Y_0) :=$`SolveMPG`(X_r, V_+, V_-, E, μ) `// solve MPG in` X_r
  $W_0 := W_0 \cup Y_0$ `// mark draws`
  for $e \in \{+, -\}$ **do**
    $F := \{(v, w) \in E \mid v \in Y_e, w \in Y_e \cup W_0\}$
    $W_0 :=$`Reach`$(W_0, -e, V_+, V_-, F)$ `// find additional draws`
    if $Y_e \cap W_0 = \varnothing$ **then** $W_e := W_e \cup Y_e$ `// mark winning positions`
  end
  $W_+ :=$`Reach`$(W_+, +, V_+, V_-, E)$
  $W_- :=$`Reach`$(W_-, -, V_+, V_-, E)$ `// solve reachability game for` (W_+, W_-)
  while $r > 0$ *and* $X_r \subset W_+ \cup W_- \cup W_0$ **do** $r := r - 1$ `// remove` X_r `if solved`
end
return (W_+, W_-, W_0)

`SolveMPG`(X, V_+, V_-, E, μ) **return** (Y_+, Y_-, Y_0)

This function finds the winning positions Y_+, Y_- and Y_0 for the mean payoff game on the subgraph of G with vertex set X and edge set $E_X = \{(v, w) \in E \mid v, w \in X\}$ with respect to the decomposition $X = (X \cap V_+) \sqcup (X \cap V_-)$ and the weight function $\mu|_X : E_X \to \mathbb{Z}$. These can be computed in $O(n^3 \cdot M_X)$ where $M_X = \max\{|\mu(v, w)| \mid (v, w) \in E_X\}$ is the size of μ_X.

`Reach`(X, e, V_+, V_-, E) **return** X'

This function computes the set of vertices X' from which player P_e can reach X, i.e. player P_e can force every play on $(V_+ \cup V_-, E)$ starting in a vertex in X' to pass through

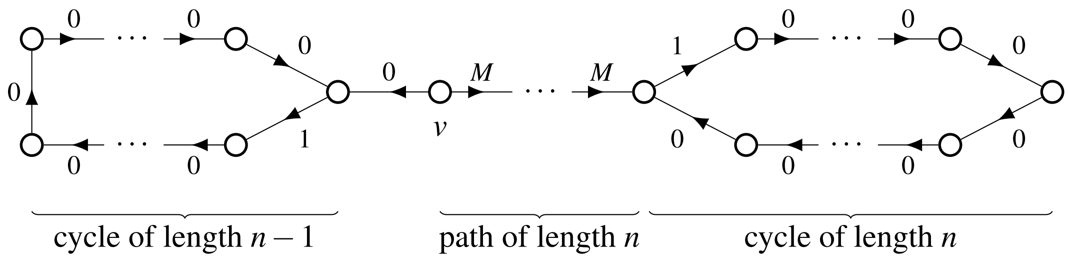


Figure 4. An example of Zwick and Paterson

a vertex in X. This can be computed in linear time in $\#E$. In fact, only edges (v,w) with $v \notin X$ and $w \in X'$ need to be considered.

We turn to the description of Algorithm 1. Roughly speaking, it computes increasing subsets W_e of $\mathbf{W}_e$ for $e \in \{+,-,0\}$ such that when it terminates, $V = W_+ \sqcup W_- \sqcup W_0$, which means that $W_e = \mathbf{W}_e$ for all $e \in \{+,-,0\}$. The algorithm begins with omitting all obsolete edges from G repeatedly until G is without obsolete edges. It initializes the variables

$$W_+ = W_- = W_0 = \varnothing, \qquad X_1 = V, \qquad \text{and} \qquad r = \begin{cases} 0 & \text{if } V = \varnothing, \\ 1 & \text{if } V \neq \varnothing. \end{cases}$$

We increase the sets W_e iteratively in the loop starting in line 10 by passing through the following steps until all winning positions are known:

(1) Decompose the complement of W in X_r into its strongly connected components and label them as $X_r, \ldots, X_{r+s-1}$ (line 12). Redefine r as $r+s-1$.
(2) Solve the mean payoff game on the subgraph with vertex set X_r, which returns winning sets Y_+, Y_- and Y_0 (line 14). Include all draw vertices of Y_0 in W_0. Test for $e \in \{+,-\}$ if player P_{-e} can reach W_0 from vertices in Y_e. If so, include these vertices in W_0; if not, then include Y_e in W_e.
(3) Solve the reachability game on G for (W_+, W_-) and include all winning positions for P_+ and P_- in W_+ and W_-, respectively (line 21 and line 22).
(4) If X_r is completely contained in W, then remove it from the list of strongly connected components (line 23) and diminish r by 1. Repeat this step until $r = 0$ or X_r is not contained in W.

Once $r = 0$, the algorithm returns (W_+, W_-, W_0).

Theorem C. *The output* (W_+, W_-, W_0) *of Algorithm 1 coincides with the winning positions* $(\mathbf{W}_+, \mathbf{W}_-, \mathbf{W}_0)$ *of the mean payoff game on G.*

Example. Zwick and Paterson consider in [4, Figure 1] a family of graphs for varying n and M, as illustrated in Figure 4, for which their algorithm requires $n^3 \cdot M$ steps to determine the characteristic value for v. Algorithm 1 computes the value $\gamma(v)$ in strictly polynomial time since it first breaks the cycles by finding obsolete edges (if there is a $-$-vertex) and then it first decomposes the graph into its strongly connected components: the minimal components are the two cycles on the left and the right, which are solved by `SolveMPG` in time $O(n^3)$, and each of the vertices w of the path connecting the cycles is an isolated component whose value $\gamma(w)$ is determined by `Reach`, which is computed in time $O(n)$.

Note that inserting additional paths between the two cycles such that the whole graph becomes strongly connected destroys the advantage of Algorithm 1 over the method of Zwick-Paterson. We exhibit in Figure 5 a family of examples (for growing n and M) of a strongly connected graph that has no obsolete edges. In this case, Algorithm 1 is essentially equal to the subroutine `SolveMPG`, which means that our method is not faster than the already existing methods to solve mean payoff games.

Further improvements. The running time of Algorithm 1 can be improved in different ways, which we describe in the following. For the sake of simplicity, we have not implemented the following strategies in Algorithm 1.

(1) All functions called by Algorithm 1 are strictly polynomial but `SolveMPG` (line 14), which makes this the most expensive subroutine in the algorithm. Since some strongly connected components decompose into smaller components during the algorithm, it makes the algorithm faster if it solves the mean payoff games on subgraphs in the order of increasing sizes of the components. This can be done most efficiently by book keeping the complete partial order on the strongly connecting components of $V - W$, and refining it whenever `StrConnComp` (line 12) is called, which allows a quick access to the components that are relatively closed in $V - W$ since they are the minimal elements with respect to the partial order.

(2) If all vertices of a component $X = X_r$ belong to one player, i.e. $X \subset V_e$ for some $e \in \{+,-\}$, then the mean payoff game on X can be solved in strictly polynomial time by Karp's algorithm; cf. [3]. More precisely, its complexity is

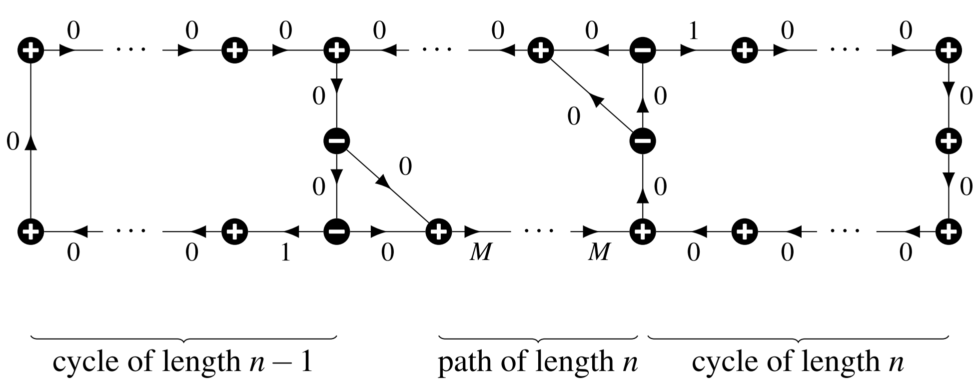


Figure 5. A mean payoff game without obsolete edges

$O(n^3)$, opposed to the complexity $O(n^3 \cdot M)$ of `SolveMPG`. Therefore it makes sense to first test whether $X_r \subset V_e$ for some $e \in \{+,-\}$, and in case running Karp's algorithm instead of `SolveMPG`.

(3) The reachability game for W_+ and W_- (line 21 and line 22) might verify unnecessarily for the same edge (v,w) multiple times if player $P_{\gamma(w)}$ can reach $W_{\gamma(w)}$ from v, in case that $v \in V_{-\gamma(w)}$ and player $P_{-\gamma(w)}$ has alternatives to playing (v,w) at v. This can be made faster by recording in an additional variable, which edges have already been considered by previous instances of `Reach`.

Finally, we comment on how our method can be used to compute the characteristic values $\chi(v)$ and to find optimal strategies. The characteristic value $\chi(v)$ of a vertex $v \in V$ can be determined by solving the mean payoff games for the altered weight functions $\mu_\lambda = \mu - \lambda$ for $\lambda \in [-M, M]$. Namely if the characteristic value of v for μ_λ is 0, then $\chi(v) = \lambda$ for μ. The value λ can be found by means of nested intervals for λ. Since $\chi(v)$ is a rational number whose denominator is at most equal to $n = \#V$, this requires $O(n \cdot \log M)$ repetitions of our algorithm. This shows that if our algorithm runs in polynomial time for all shifted weights μ_λ, then we can determine $\chi(v)$ in polynomial time. In particular, if μ has the signature of a potential, then we find $\chi(v)$ for all $v \in V$ in $O(n^3 \cdot \log M)$ steps.

Note that the deduction of an optimal strategy σ_e is more sublte. It must satisfy $\chi(\sigma_e(v)) = \chi(v)$ for every $v \in V_e^*$. If the game is sufficiently generic, i.e. if there are no distinct cycles with the same mean payoff, then any choice of $\sigma_e(v)$ with $\chi(\sigma_e(v)) = \chi(v)$ for $v \in V_e^*$ yields an optimal strategy. But in general, the choices of $\sigma_e(v)$ are not independent.

1. Proof of Proposition B

We begin with the proof of Proposition B because we apply it (in the case of 2-obsolete edges) to prove Theorem A.

Lemma 1.1. *Let k be a positive integer and $(v,w) \in E$ an k-obsolete edge that appears in an optimal game. Then $\gamma(v) = -\epsilon(v)$.*

Proof. Let $\pi = (v_1, \dots)$ be an optimal game that starts at $v_1 = v$ with $v_2 = w$, following optimal strategies σ_+ and σ_- for players P_+ and P_-, respectively. Let $e = \epsilon(v)$. Since (v,w) is k-obsolete, player P_{-e} has a strategy σ'_{-e} such that the game $\pi' = (v'_0, \dots)$ following the strategies σ_e and σ'_{-e} and starting at $v'_0 = v$ (and thus $v_1 = w$) ends in a sink of P_e or passes through v again, i.e. π' follows periodically a cycle $(v'_0, \dots, v'_l)$ with $v'_l = v'_0 = v$ for some $l \leqslant k$, and such that

$$e \cdot l \cdot \chi(\pi') \;=\; e \cdot \Big(\sum_{i=0}^{l-1} \mu(v_i, v_{i+1}) \Big) \;<\; 0.$$

If π' ends in a sink, then P_{-e} wins, and $\gamma(v) = -e$ as claimed. If π' returns to v'_0, then we conclude that

$$-e\chi(\pi) \;\geqslant\; -e\chi(\pi') \;>\; 0$$

Algorithm 2: Find all k-obsolete edges in G

Input: k, V_+, V_-, E, μ
Output: F

Function `Obsolete`(k, V_+, V_-, E, μ) **is**
  $F = \varnothing$
  for $e \in \{+,-\}$ *and* $v \in V_e$ **do**
    for $i = 0, \dots, k-1$ **do** $\eta_i(v) = 0$ `// initial values for` $\eta_i(v)$
    for $w \in (V_+ \cup V_-) - \{v\}$ **do** $\eta_0(w) := e \cdot \infty$ `// and` $\eta_0(w)$
    for $i = 1, \dots, k-1$ **do**
      for $w \in (V_+ \cup V_-) - \{v\}$ **do**
        $\eta_i(w) := \epsilon(w) \cdot \max\{\epsilon(w) \cdot (\mu(w,u) + \eta_{i-1}(u)) \mid (w,u) \in E\}$
      end   `// iterative computation of` $\eta_i(w)$
    end
    for $(v,w) \in E$ **do**
      if $e \cdot \eta_{k-1}(w) < -e \cdot \mu(v,w)$ **then** $F := F \cup \{(v,w)\}$
    end   `// find and mark` k`-obsolete edges`
  end
  return F
end

since π is $-e$-optimal, and therefore $\gamma(v) = \operatorname{sign}\big(\chi(\pi)\big) = -e$. □

Proof of Proposition B. We begin with the proof that the winning positions do not change if we omit the k-obsolete edges from E. Consider $v_0 \in V$ and choose a game $\pi = (v_0, \dots)$ following optimal strategies σ_+ and σ_- of P_+ and P_-, respectively. If π does not contain any k-obsolete edge, then it also optimal for the mean payoff game that results from G when we omit the k-obsolete edges. Thus $\chi(v_0)$ and $\gamma(v_0)$ do not change in this case.

If π passes through a k-obsolete edge (v,w), i.e. $v = v_i$ and $w = v_{i+1}$ for some $i \geqslant 0$, then player $P_{-\epsilon(v)}$ can achieve that a game $\pi' = (v_i, v_{i+1}, \dots,)$ starting in $v_i = v$ and following $\sigma_{\epsilon(v)}$ either ends in a sink of $P_{\epsilon(v)}$ or returns to v in $l \leqslant k$ moves, i.e. $v_{i+l} = v_i = v$, such that $\epsilon(v)\mu(\pi') < 0$. If π' ends in a sink, then $\epsilon(v)\chi(v_0) = -\infty < 0$; otherwise we have

$$\epsilon(v)\chi(v_0) \;=\; \epsilon\chi(v) \;\leqslant\; \epsilon(v)\chi(\pi') \;<\; 0$$

since π' is $\epsilon(v)$-optimal and since $\chi(v_0) = \chi(\pi) = \chi(v)$.

This and Lemma 1.1 show that that $\epsilon(v_i) = -\gamma(v_i) = -\gamma(v_0)$ whenever (v_i, v_{i+1}) is a k-obsolete edge that occurs in π. Thus omitting the k-obsolete edges from G allows player $P_{-\epsilon(v_0)}$ to use the same optimal strategy at vertices occurring in π, which means that an $-\epsilon(v)$-optimal play π'' starting in v_0 for the mean payoff game without the k-obsolete edges is also $-\epsilon(v)$-optimal for G. Thus $\epsilon(v)\chi(\pi'') \leqslant \epsilon(v)\chi(\pi) < 0$ and $\gamma(v_0)$

does not change if we omit the k-obsolete edges from G. This shows that the winning positions of G do not change if we omit the k-optimal edges.

We continue with the proof that the output F of Algorithm 2 coincides with the set of all k-obsolete edges. To describe the algorithm in words, it defines for choice of a fixed vertex $v \in V$ and $e = \epsilon(v)$, a function $\eta_i(w)$ in $i = 0, \dots, k-1$ and $w \in V$ that is given by $\eta_i(v) = 0$ for all $i = 0, \dots, k-1$, by $\eta_0(w) = e \cdot \infty$ for all $w \neq v$ and recursively by

$$\eta_i(w) \ = \ \epsilon(w) \cdot \max\{\epsilon(w) \cdot (\mu(w,u) + \eta_{i-1}(u)) \mid (w,u) \in E\}$$

for all $w \neq v$ and $i = 1, \dots, k-1$. It defines F as the set of all edges (v,w) for which

$$e \cdot \eta_{k-1}(w) \ < \ -e \cdot \mu(v,w).$$

In order to prove that F coincides with the set of k-obsolete edges of G, we consider the following game on G: for fixed v_0 and i, a play consists of a finite directed path $\pi = (v_0, \dots, v_i)$ of length i unless it runs into a sink v_j and ends before; its value is $\eta(\pi) = -\epsilon(v_j) \cdot \infty$ if it ends in a sink v_j, and otherwise it is $\eta(\pi) = e \cdot \infty$ if $v \notin \{v_0, \dots, v_i\}$ and

$$\eta(\pi) \ = \ \sum_{j=0}^{l-1} \mu(v_j, v_{j+1})$$

if $v \in \{v_0, \dots, v_i\}$ where $l = \min\{j \in \mathbb{N} \mid v_j = v\}$. Then an edge (v,w) is k-obsolete if and only if player P_{-e} can force every play π of length $k-1$ starting at w to have value $e \cdot \eta(\pi) < -e \cdot \mu(v,w)$. Thus it follows that F is the set of k-obsolete edges once we can show that $\eta(\pi) = \eta_i(w)$ for every play of length i starting at w that is played under optimal strategies for both players.

If $w = v$, then we have $v_0 = v$ and thus $l = 0$, which shows that $\eta(\pi) = 0 = \eta_i(v)$ for every $i = 0, \dots, k-1$ and every play π starting at $w = v$. If $w \neq v$, then we show the claim by induction on i. For $i = 0$, we have $\eta(\pi) = e \cdot \infty = \eta_0(w)$.

For $i > 0$, consider an optimal play π with first move $(w,u) = (v_0, v_1)$. Using the inductive hypothesis on the play $\pi' = (v_1, \dots, v_i)$ of length $i-1$, we have

$$\eta(\pi) \ = \ \mu(w,u) + \eta(\pi') \ = \ \mu(w,u) + \eta_{i-1}(u).$$

Since player $P_{\epsilon(w)}$ aims at maximizing $\epsilon(w)\eta(\pi)$, we have

$$\epsilon(w)\eta(\pi) \ = \ \max\left\{\epsilon(w)\big(\mu(w,u) + \eta_{i-1}(u)\big) \,\middle|\, (w,u) \in E\right\} \ = \ \epsilon(w)\eta_i(w),$$

and thus $\eta_i(w) = \eta(\pi)$, as claimed. This shows that the output F of Algorithm 2 is precisely the set of all k-obsolete edges of G. It is evident from the architecture of Algorithm 2 that its complexity is in $O(k \cdot n^2)$. □

2. Proof of Theorem A

As a first step towards a proof of Theorem A, we reason that we can assume that the edge set E of G coincides with the edge set E' of the derived reachability game of G.

To wit, E' results from omitting all 2-obsolete edges from E, which are the edges $(v,w) \in E$ for which $\epsilon(w) = -\epsilon(v)$ and $\epsilon(v)\big(\mu(v,w) + \mu(w,v)\big) < 0$. As a consequence of Proposition B, the omittance of all 2-obsolete edges does not change the winning positions of G.

Note that whenever both (v,w) and (w,v) are in E' and $\epsilon(v) \neq \epsilon(w)$, then $\mu(v,w) + \mu(w,v) = 0$. Moreover, if μ has the signature of a potential ψ, i.e.

$$\text{(P0)} \qquad \mu(v,w) \;=\; \mu(w,v) \;=\; \psi(v)-\psi(w) \;=\; 0 \qquad \text{if } \epsilon(v)=\epsilon(w),$$

$$\text{(P1)} \qquad \operatorname{sign}\big(\epsilon(v)\mu(v,w)-\epsilon(w)\mu(w,v)\big) \;=\; \operatorname{sign}\big(\psi(v)-\psi(w)\big) \qquad \text{if } \epsilon(v)\neq\epsilon(w)$$

for all $(v,w) \in E$, then either condition remains unaltered for the restriction μ' of μ to E' (which we extend by $\mu'(v,w) = -\epsilon(v)\infty$ to all of $V \times V$): this is obvious for (P0) and it follows for (P1) since if $(v,w) \in E'$ and $(w,v) \in E - E'$, i.e. $\mu'(w,v) = -\epsilon(w)\infty$, then $\epsilon(w) = -\epsilon(v)$ and $\epsilon(w)\big(\mu(w,v)+\mu(v,w)\big) < 0$, which implies $\operatorname{sign}\big(\mu(v,w)+\mu(w,v)\big) = -\epsilon(w)$ and thus

$$\begin{aligned} \operatorname{sign}\big(\epsilon(v)\mu(v,w)-\epsilon(w)(-\epsilon(w)\infty)\big) \;&=\; 1 \;=\; \epsilon(v)\operatorname{sign}\big(\mu(v,w)+\mu(w,v)\big) \\ &=\; \operatorname{sign}\big(\epsilon(v)\mu(v,w)-\epsilon(w)\mu(w,v)\big) \;=\; \operatorname{sign}\big(\psi(v)-\psi(w)\big). \end{aligned}$$

This shows that if μ has the signature of a potential ψ, then μ' does so too. To conclude, we can and will assume that $E = E'$ for the purpose of this proof.

Lemma 2.1. *Assume that $E = E'$ and that μ has the signature of a potential ψ. Let $X \subset V$ be a strongly connected subset. Then for every $(v,w) \in E$ with $v,w \in X$, we have*

$$\begin{aligned} \mu(w,v) &= 0 && \textit{if } \epsilon(v)=\epsilon(w), \\ \mu(v,w)+\mu(w,v) &= 0 && \textit{if } \epsilon(v)\neq\epsilon(w). \end{aligned}$$

Proof. Let $(v,w) \in E$ with $v,w \in X$. In the case $\epsilon(v) = \epsilon(w)$, the claim $\mu(v,w) = 0$ follows at once from the hypothesis that μ has the signature of a potential. If $\epsilon(v) \neq \epsilon(w)$, then (P1) implies that

$$\operatorname{sign}\big(\psi(v)-\psi(w)\big) \;=\; \epsilon(v)\big(\mu(v,w)+\mu(w,v)\big) \;\geqslant\; 0,$$

and thus $\psi(v) \geqslant \psi(w)$, which is also true if $\epsilon(v) = \epsilon(w)$ by (P0). Since X is strongly connected, there is a path $(v_1,\dots,v_n)$ from $v_1 = w$ to $v_n = v$ with $v_1,\dots,v_n \in X$. Concatenation with (v,w) yields a closed path $(v_0,v_1,\dots,v_n)$ from $v_0 = v$ to v. By what we have proven,

$$\psi(v) \;=\; \psi(v_0) \;\geqslant\; \psi(v_1) \;\geqslant\; \dots \;\geqslant\; \psi(v_n) \;=\; \psi(v),$$

which implies equality throughout. Thus

$$\operatorname{sign}\big(\mu(v,w)+\mu(w,v)\big) \;=\; \epsilon(v)\operatorname{sign}\big(\psi(v)-\psi(w)\big) \;=\; 0,$$

which establishes our claim $\mu(v,w)+\mu(w,v) = 0$ in the case $\epsilon(v) \neq \epsilon(w)$. □

Proof of Theorem A. Assume that $E = E'$ and that μ has the signature of a potential. Let $(\mathbf{W}_+,\mathbf{W}_-,\mathbf{W}_0)$ be the winning positions for the mean payoff game on G. Let $(\mathbf{Y}_+,\mathbf{Y}_-,\mathbf{Y}_0)$ be the winning positions for the reachability game on G' with respect to the sets $\mathbf{Z}_+$ and $\mathbf{Z}_-$ of sinks in V_- and V_+, respectively. Let σ_+ and σ_- be optimal strategies for the mean payoff game on G. We aim to show that $\mathbf{W}_e = \mathbf{Y}_e$ for all $e \in \{+,-,0\}$.

Consider $e \in \{+,-\}$ and $v \in \mathbf{Y}_e$. Then player P_e can reach $\mathbf{Z}_e$ from v, i.e. the optimal play π following σ_+ and σ_- that starts in v is finite and ends in a sink $w \in V_{-e}$. Thus P_e wins the mean payoff game on G and $v \in \mathbf{W}_e$.

If $v \in \mathbf{Y}_0$, then the optimal game $\pi = (v,\dots)$ following σ_+ and σ_- does reach neither $\mathbf{Z}_+$ nor $\mathbf{Z}_-$, and is therefore infinite. This means that π is eventually periodic. By Lemma 2.1, the payoff of a period is 0, which shows that $\chi(\pi) = 0$ and therefore $v \in \mathbf{W}_0$. This shows that $\mathbf{W}_e = \mathbf{Y}_e$ for all $e \in \{+,-,0\}$, as desired.

It is well-known that reachability games are strictly linear in $\#V + \#E$. For instance, we can use the strictly linear Algorithm 5 to determine $\mathbf{Y}_+ = \texttt{Reach}(\mathbf{Z}_+, V_+, V_-, E)$, $\mathbf{Y}_- = \texttt{Reach}(\mathbf{Z}_-, V_+, V_-, E)$ and $\mathbf{Y}_0 = V - (\mathbf{Y}_+ \cup \mathbf{Y}_-)$. □

3. Proof of Theorem C

The aim of this section is a proof of Theorem C, i.e. a proof of correctness for Algorithm 1. We will derive this proof from a number of auxiliary results.

Definition 3.1. Let $W \subset Y \subset V$ and $e \in \{+,-\}$. We say that player P_e *reaches* W *within* Y *from* v if he or she has a strategy such that every play with all vertices in Y following this strategy passes through a vertex in W (where a play terminates once it reaches a vertex in W). If $Y = V$, then we simply say that player P_e *reaches* W *from* v. We say that W is *e-reachably saturated* if every vertex v from which player P_e reaches W belongs to v. We say that a triple (W_+, W_-, W_0) of subsets of V is *reachably saturated* if W_+ is $+$-reachably saturated and if W_- is $-$-reachably saturated.

Remark 3.2. More formally, we can express the definition of an e-reachably saturated subset W of V as follows: W is e-reachably saturated if and only if for every edge $(v,w) \in E$ with $w \in W$ also $v \in W$ if either $v \in V_e$ or if for every other edge $(v,w') \in E$ also $w' \in W$.

Definition 3.3. Let $W \subset V$ and $X \subset V - W$. We say that X is *relatively closed in* $V - W$ if for every $v \in X$ and every $(v,w) \in W$ also $w \in X$. If $W = \varnothing$, then we simply say that X is *(successor) closed*.

Remark 3.4. Note that the closed subsets of V form the closed subsets of a topology on V and that a subset X of $V - W$ is relatively closed if and only if it is closed in the subspace topology of $V - W$. Note further that $V - W$ is closed if and only if W is both $+$-reachably and $-$-reachably saturated.

For a subset X, we let $X_+ = X \cap V_+$, $X_- = X \cap V_-$, $E_X = \{(v,w) \in E \mid v,w \in X\}$ and $\mu_X : X \times X \to \mathbb{Z} \cup \{\pm\infty\}$ be the respective restrictions to X. We denote the characteristic value of a vertex $v \in X$ with respect to the mean payoff game on X, as given by X_+, X_-, E_X and μ_X, by $\chi_X(v)$ and define $\gamma_X(v) = \operatorname{sign}\big(\chi_X(v)\big)$.

Lemma 3.5. *Let $W_e \subset \mathbf{W}_e$ be subsets for $e \in \{+,-,0\}$ such that (W_+, W_-, W_0) is reachably saturated and let X be a relatively closed in $V - W$ where $W = W_+ \cup W_- \cup W_0$. Let $v \in X$. If $\gamma_X(v) = 0$, then $\gamma(v) = 0$.*

Proof. Let $\gamma_X(v) = 0$ and consider an optimal play $\pi = (v_1, \dots)$ for the mean payoff game on V that starts in $v_1 = v$. If all vertices of π are contained in X, then π is also an optimal play for the mean payoff game on X. Thus $\chi(v) = \chi(\pi) = \chi_X(v) = 0$ and $\gamma(v) = 0$, as claimed.

If π contains a vertex that is not in X, then we let v_{n+1} be the first such vertex; in particular $v_n \in X$. Since X is relatively closed in $V - W$, we conclude that $v_{n+1} \in W = W_+ \cup W_- \cup W_0$. Since $W_{\epsilon(v_n)}$ is $\epsilon(v_n)$-reachably saturated and $v_n \notin W_{\epsilon(v_n)}$, neither v_{n+1} is in $W_{\epsilon(v_n)}$.

Since $\gamma(v_{n+1}) \neq \epsilon(v_n)$ and

$$0 \;=\; \epsilon(v_n) \cdot \gamma_X(v_n) \;\leqslant\; \epsilon(v_n) \cdot \gamma(v_n) \;=\; \epsilon(v_n) \cdot \gamma(v_{n+1}),$$

we conclude that $\gamma(v) = \gamma(v_{n+1}) = 0$, as desired. □

Lemma 3.6. *Let $W_e \subset \mathbf{W}_e$ be subsets for $e \in \{+,-,0\}$ such that (W_+, W_-, W_0) is reachably saturated and let X be a relatively closed in $V - W$ where $W = W_+ \cup W_- \cup W_0$. Let $v \in V$ and $\gamma_X(v) \neq 0$. Let Y_+, Y_- and Y_0 be the winning positions for the mean payoff game on X. If player $P_{-\gamma_X(v)}$ reaches W_0 within $Y_{\gamma_X(v)} \cup W_0$ from v, then $\gamma(v) = 0$.*

Proof. Let $e = -\gamma_X(v)$. Then $e\gamma_X(v) = -e^2 < 0$. Since player P_e reaches W_0 within $Y_{\gamma_X(v)} \cup W_0$ from v, we have $e\gamma(v) \geqslant 0$ and thus $\gamma_X(v) \neq \gamma(v)$. Let $\pi = (v_1, \dots)$ be an optimal play in V that starts in $v_1 = v$. Since $\gamma_X(v) \neq \gamma(v)$ and an optimal play for the mean payoff game on X stays in Y_{-e}, there must a vertex v_{n+1} that is not contained in Y_{-e}.

Assuming that n is minimal and thus $v_n \in X$, we conclude that $v_{n+1} \in W \cup Y_e \cup Y_0$ since X is relatively closed in $V - W$. Since P_e reaches W_0 and π is e-optimal, $v_{n+1} \notin W_{-e}$. Since W_e is e-reachably saturated and π is $-e$-optimal, $v_{n+1} \notin W_e$. Since $\gamma_X(v) = -e$, $v_{n+1} \notin Y_e$. Thus $\gamma(v) \neq e = -\gamma_X(v)$ and therefore $\gamma(v) = 0$, as claimed. □

Corollary 3.7. *Let $W_e \subset \mathbf{W}_e$ be subsets for $e \in \{+,-,0\}$ such that (W_+, W_-, W_0) is reachably saturated and let X be a relatively closed in $V - W$ where $W = W_+ \cup W_- \cup W_0$. Let Y_+, Y_- and Y_0 be the winning positions for the mean payoff game on X. Then*

$$Y_0 \,\cup\, \Big\{ v \in Y_e \,\Big|\, e \in \{+,-\} \text{ and } P_{-e} \text{ can reach } W_0 \text{ within } Y_e \cup W_0 \text{ from } v \Big\} \;\subset\; \mathbf{W}_0$$

and if player P_{-e} cannot reach W_0 from any vertex of Y_e, then $Y_e \subset \mathbf{W}_e$ for $e \in \{+,-\}$.

Proof. By Lemma 3.5, we have $Y_0 \subset \mathbf{W}_0$. If $v \in Y_{-e}$ and P_e can reach W_0 within $Y_e \cup W_0$ from v, then $v \in \mathbf{W}_0$ by Lemma 3.6. This the first claim.

Assume that W_0 is not $-e$-reachable within $Y_e \cup W_0$ from any vertex $v \in Y_e$. Let $\pi = (v_1, \dots)$ be an optimal play starting in $v = v_1$. If all vertices of π are contained in Y_e, then π is optimal for the mean payoff game in X, and thus $\gamma(v) = \gamma(\pi) = \gamma_X(v) = e$.

If not, then there is a vertex v_{n+1} that is not contained in X. We assume that n is minimal and thus $v_n \in Y$. Since X is relatively closed in $V - W$, we must have $v_{n+1} \in W$.

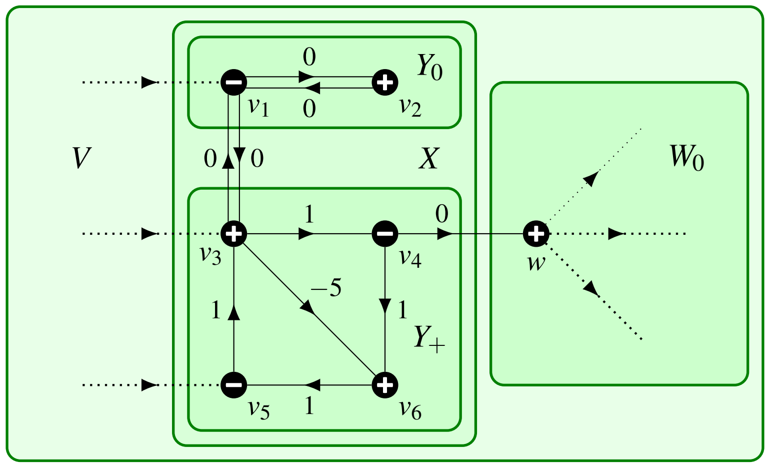


Figure 6. Additional draw positions in a relatively closed subset X

Since neither W_{-e} nor W_0 is $-e$-reachable by our assumptions, $v_{n+1} \in W_e$ and thus $\gamma(v) = \gamma(\pi) = e$, as desired. □

Example 3.8. Figure 6 illustrates an example for the situation of Lemma 3.6 and Corollary 3.7, in which a difference between $\gamma_X(v)$ and $\gamma(v)$ occurs. Namely, the mean payoff game on the relatively closed subset X of $V - W$ has the winning sets

$$Y_0 = \{v_1, v_2\}, \qquad Y_+ = \{v_3, v_4, v_5, v_6\}, \qquad Y_- = \varnothing,$$

and thus $\gamma_X(v_i) = +$ for $i = 3, \ldots, 6$. An optimal game within X that passes through these vertices is $\pi = (v_3, v_4, v_6, v_5, v_3, \ldots)$, and it has positive mean payoff $\chi(\pi) = 1$. However, player P_- can reach W_0 from v_4 (within $Y_+ \cup W_0$) by playing the edge (v_4, w) and achieve a draw play in this way, which shows that $\gamma(v_4) = 0$. We conclude that π is not optimal V. The only play in Y_+ that avoids the vertex v_4 has negative mean payoff $\chi(v_3, v_6, v_5, \ldots) = -1$, which implies that also $\gamma(v_3) = \gamma(v_5) = \gamma(v_6) = 0$.

With this, we are prepared to prove the central result of the paper.

Proof of Theorem C. We begin with the initial steps of Algorithm 1. By Proposition B, we can omit all obsolete edges from E, which justifies the replacement of E by the complement of all such edges in line 3 of the algorithm. The next steps define the initial values $W_+ = W_- = W_0 = \varnothing$, $r = 1$ and $X_1 = V$, which satisfy the following properties:

(1) $W_e \subset \mathbf{W}_e$ for all $e \in \{+, -, 0\}$;
(2) (W_+, W_-, W_0) is reachably saturated;
(3) $V - W \subset X_1 \cup \ldots \cup X_r$;
(4) if there is an oriented path from a vertex in X_i to a vertex in X_j, then $i \leqslant j$;
(5) if $r > 0$, then $X_r \not\subset W$.

Assuming the validity of (1)–(5), we prove in the following that every iteration of the loop in line 10 changes the variables W_+, W_-, W_0, r, $X_1, \ldots, X_r$ in a way such that the new values again satisfy (1)–(5) and such that $W = W_+ \cup W_- \cup W_0$ increases by at least one element.

Note that property (3) implies that if $r = 0$, then $V - W \subset \varnothing$, and thus $W = V$. This reasons that once the condition $r > 0$ is not valid and the algorithm returns (W_+, W_-, W_0), then $W_e = \mathbf{W}_e$ for all $e \in \{+,-,0\}$ by (1). Conversely, if $W = V$, then $X_r \subset W$ for all $r > 0$ and thus $r = 0$ by (5). Since W strictly increases with every iteration of the loop, this shows that the algorithm terminates after finitely many steps.

We are left with proving that (1)–(5) hold after each iteration of the loop starting in line 10. For the purpose of this proof, we mark the new values of W_+, W_-, W_0, r and $X_1, \ldots, X_r$ at the end of an iteration with a prime, i.e. we denote them by W'_+, W'_-, W'_0, r' and $X'_1, \ldots, X'_{r'}$. We begin with inspecting the effect of the individual steps of loop on these variables before verifying the conditions (1)–(5).

Line 11–line 13: These steps define $(s, X'_r, \ldots, X'_{r+s-1}) = \texttt{StrConnComp}(X_r - W, E)$ and $r'' = r + s - 1$, which satisfy that $X_r - W = X'_r \cup \ldots \cup X'_{r+s-1}$. Setting $X'_i = X_i$ for $i = 1, \ldots, r-1$, we conclude from (3) that

$$V - W \ \subset \ (X_1 \cup \ldots \cup X_r) - W \ = \ (X'_1 \cup \ldots \cup X'_{r-1} \cup X'_r \cup \ldots \cup X'_{r''}) - W.$$

Since there is no path from X_r to X_i for $i < r$ by (4) and no path from $X'_{r''}$ to X'_j for $j = r, \ldots, r'' - 1$, we conclude that $X'_{r''}$ is relatively closed in $V - W$. Note that $X'_{r''} \subset X_r - W$ has empty intersection with W.

Line 14–line 19: These steps define $(Y_+, Y_-, Y_0) = \texttt{SolveMPG}(X'_{r''}, V_+, V_-, E, \mu)$, which are the winning sets of the mean payoff game on the subgame of G with vertex set $X'_{r''}$, as well as

$$W''_0 \ = \ W_0 \cup Y_0 \cup \left\{ v \in Y_e \,\middle|\, e \in \{+,-\} \text{ and } P_{-e} \text{ can reach } W_0 \text{ within } Y_e \cup W_0 \text{ from } v \right\}$$

and for $e \in \{+,-\}$, $W''_e = W_e \cup Y_e$ if $Y_e \cap W''_0 = \varnothing$ and $W''_e = W_e$ otherwise. Since (W_+, W_-, W_0) is reachably saturated by (2) and $X'_{r''}$ is relatively closed in $V - W$, Corollary 3.7 shows that $W''_e \subset \mathbf{W}_e$ and thus (1) holds for W''_e for all $e \in \{+,-,0\}$.

Line 21–line 22: These steps define $W'_e = \texttt{Reach}(W''_e, V_+, V_-, E)$ as the set of vertices from which player P_e can reach W''_e for $e \in \{+,-\}$. In particular, W'_e is e-reachably saturated for $e \in \{+,-\}$. We define $W'_0 = W''_0$.

Line 23: This last step of the loop defines r' as the largest positive integer $k \leqslant r''$ such that $X'_k \not\subset W'_+ \cup W'_- \cup W'_0$ if there is any such k, and as $r' = 0$ otherwise.

We turn to the verification of (1)–(5) for W'_+, W'_-, W'_0, r' and $X'_1, \ldots, X'_{r'}$. Since $W''_e \subset \mathbf{W}_e$ and player P_e can reach W''_e from all vertices in W'_e, we have $W'_e \subset \mathbf{W}_e$ for $e \in \{+,-\}$. Moreover $W'_0 = W''_0 \subset \mathbf{W}_0$, which shows (1). Since W'_e is e-reachably saturated for $e \in \{+,-\}$, we have (2). Since $X'_{r'+1} \cup \ldots \cup X'_{r''} \subset W$, we have

$$V - W \ \subset \ (X'_1 \cup \ldots \cup X'_{r''}) - W \ \subset \ X'_1 \cup \ldots \cup X'_{r'},$$

which verifies (3). Since (4) holds for $X_1, \ldots, X_r$, there is no path from $X'_j = X_j$ to $X'_i = X_i$ for $1 \leqslant i < j < r$. By the properties of $\texttt{StrConnComp}$, there is no path from X'_j to X'_i for

$r \leqslant i < j \leqslant r'$. Since $X'_j \subset X_r$ for $r \leqslant j \leqslant r'$, we conclude that there is no path from X'_j to X'_i for $1 \leqslant i < r \leqslant j \leqslant r'$. Thus (4) for $X'_1, \dots, X'_{r'}$. Since $X'_{r'} \not\subset W$ if $r' > 0$, we have (5). This completes the proof of Theorem C. □

Appendix A. Pseudo-code of all functions

In this appendix, we include pseudo code, together with a short description, for the functions `StrConnComp`, `SolveMPG` and `Reach`, which are called in Algorithm 1.

`StrConnComp`(X,E) **return** $(s, Y_s, \dots, Y_1)$

The idea for finding the strongly connected components of X with respect to the edge set $E_X = \{(v,w) \mid v,w \in X\}$ is to choose an arbitrary path in X with the property that it runs along new edges as long as this is possible. Once no new edges can be found, then the path must circle in a minimal strongly connected component, which we exclude from X. An iteration of this strategy yields all strongly connected components of X.

Technically this is done as follows in Algorithm 3. We choose a random path $\pi = (v_1, \dots, v_i)$ in X (with i increasing) that picks a new edge whenever possible. If a vertex v_i occurs for a second time in π, i.e. $v_i = v_j$ for some $j < i$, then we this means that all vertices $v_j, \dots, v_i$ belong to the same strongly connected component. We mark the smallest (known) index j such that there is a path from v_i to v_j by m_i and search for new outgoing edges from all the vertices $v_{m_i}, \dots, v_i$. Once there are no outgoing edges anymore, then we know that $v_{m_1}, \dots, v_i$ is a strongly connected component that does not allow a path to any other connected component, which we call Y_1. We exclude $v_{m_i}, \dots, v_i$ from X and from π, replace i by $m_i - 1$ and continue with the procedure. In this way we find the strongly connected components $Y_1, \dots, Y_s$ of X with the property that if there is a path from Y_i to Y_j, then $i < j$. Note that for the desired outcome for Algorithm 1, we have to reverse the order of $Y_1, \dots, Y_s$.

`SolveMPG`(X, V_+, V_-, E, μ) **return** (Y_+, Y_-, Y_0)

We implement Zwick-Paterson's method from [4] to solve the mean payoff game on X, under the assumption that X is strongly connected. The idea is to compute the payoff $\eta(v) = \eta_k(\pi)$ of optimal games $\pi = (v_0, \dots, v_k)$ of fixed length k starting at $v = v_0$. The estimate

$$k \cdot \chi(v) - 2nM \;\leqslant\; \eta_k(v) \;\leqslant\; k \cdot \chi(v) + 2nM$$

from [4, Thm. 2.2] implies that if $\eta_k(v) > 2nM$, then $\gamma_X(v) = +$ and that if $\eta_k(v) < -2nM$, then $\gamma_X(v) = -$. By [4, Thm. 2.4], we have $\gamma_X(v) = 0$ if $-2nM \leqslant \eta_k(v) \leqslant 2nM$ for $k = 4n^2M + 1$.

Our implementation in Algorithm 4 checks first if the strongly connected component X has contains any edges. If not, then X consists of a single sink v, which yields $\gamma_X(v) = -\epsilon(v)$. If not, then it computes recursively the values $\eta_k(v)$ (for increasing k) and tests whether $\eta_k(v) > 2nM$ or $\eta_k(v) < -2nM$. After $4n^2M + 1$ steps it marks all vertices that are not identified as winning positions of P_+ or P_- as draw positions.

Algorithm 3: Find all strongly connected components of a subgraph

Input: X, E
Output: $s, Y_s, \ldots, Y_1$

```
Function StrConnComp(X, E) is
    s := 0
    i := 0
    F := {(v,w) ∈ E | v,w ∈ X}                    // initialize the variables
    while Y_1 ∪ ... ∪ Y_s ≠ X do
        if i = 0 then                              // start a list of vertices if i = 0
            i := 1
            choose any v_1 ∈ X − (Y_1 ∪ ..., Y_s)
            m_1 := 1
            j := 1
        end
        while j ⩽ i do                             // if m_i ⩽ j ⩽ i, then v_j is reachable from v_i
            for (v_j, w) ∈ F do                    // search for w reachable from v_j
                F := F − {(v_j, w)}
                if w = v_k for some k < m_i then   // if v_k reachable from v_j,
                    for l = k+1, ..., i do m_l := m_k   // then for all l = k, ..., i,
                    j := m_k                       // v_{m_k} is reachable from v_l
                end
                else if w ∉ {v_{m_i}, ..., v_i} then   // new entry w in the list
                    i := i + 1
                    v_i := w
                    m_i := i
                    j := i
                end
            end
            j := j + 1                             // if ¬∃(v_j, w) ∈ F, then increase j
        end                                        // there is no edge leaving v_{m_i}, ..., v_i if j > i
        s := s + 1
        Y_s := {v_{m_i}, ..., v_i}                 // define the new component Y_s
        i := m_i − 1
        j := m_i
    end
    return (s, Y_s, ..., Y_1)
end
```

Algorithm 4: Solving the mean payoff game on the subgraph X

```
Input: $X, V_+, V_-, E, \mu$
Output: $Y_+, Y_-, Y_0$
Function SolveMPG($X, V_+, V_-, E, \mu$) is
    $Y_+ = Y_- = Y_0 := \varnothing$
    $C := \{(v,w) \in E \mid v, w \in X\}$
    if $C = \varnothing$ and $X \subset V_+$ then $Y_- := X$
    if $C = \varnothing$ and $X \subset V_-$ then $Y_+ := X$
    if $C \neq \varnothing$ then
        $n := \#V_+ + \#V_-$
        $M := \max\{|\mu(v,w)| \mid (v,w) \in C\}$
        for $v \in X$ do $\eta_0(v) := 0$
        for $k = 1, \dots, 4n^2M + 1$ do
            for $v \in X$ do
                if $v \in V_+$ then $\eta_k(v) := \max\{\mu(v,w) + \eta_{k-1}(w) \mid (v,w) \in C\}$
                if $v \in V_-$ then $\eta_k(v) := \min\{\mu(v,w) + \eta_{k-1}(w) \mid (v,w) \in C\}$
                if $\eta_k(v) > 2nM$ then $Y_+ := Y_+ \cup \{v\}$
                if $\eta_k(v) < -2nM$ then $Y_- := Y_- \cup \{v\}$
            end
        end
        $Y_0 := X - (Y_+ \cup Y_-)$
    end
    return $(Y_+, Y_-, Y_0)$
end
```

Algorithm 5: Finding all vertices from which player P_e can reach X

```
Input: $X, e, V_+, V_-, E$
Output: $X$
Function Reach($X, e, V_+, V_-, E$) is
    for $(v,w) \in E$ with $v \notin X$ and $w \in X$ do
        if $v \in V_e$ or $u \in X$ for all $(v,u) \in E$ then
            $X := X \cup \{v\}$
        end
    end
    return $X$
end
```

`Reach`(X, e, V_+, V_-, E) **return** X'

We find the set X' of all vertices from which player P_e can reach X by the following recursive procedure in Algorithm 5. We begin with $X' := X$. Whenever there is an edge $(v,w) \in E$ such that $w \in X'$, then we include v in X' if either $\epsilon(v) = e$ or if $w' \in X'$ for all edges $(v,w') \in E$. Once there is no more edge with one of these properties, the function returns X'.

References

[1] Dietmar Berwanger and Olivier Serre. Parity games on undirected graphs. *Inform. Process. Lett.*, 112(23):928–932, 2012.

[2] Andrzej Ehrenfeucht and Jan Mycielski. Positional strategies for mean payoff games. *Internat. J. Game Theory*, 8(2):109–113, 1979.

[3] Richard M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Math.*, 23(3):309–311, 1978.

[4] Uri Zwick and Mike Paterson. The complexity of mean payoff games on graphs. *Theoret. Comput. Sci.*, 158(1-2):343–359, 1996.

Marianne Akian, École Polytechnique, Paris, France
Email address: `marianne.akian@inria.fr`

Stéphane Gaubert, École Polytechnique, Paris, France
Email address: `stephane.gaubert@inria.fr`

Oliver Lorscheid, Instituto Nacional de Matemática Pura e Aplicada, Rio de Janeiro, Brazil
Email address: `oliver@impa.br`

Matthias Mnich, Technische Universität Hamburg, Hamburg, Germany
Email address: `algo@tuhh.de`