ISSN 0965-5425, Computational Mathematics and Mathematical Physics, 2016, Vol. 56, No. 4, pp. 688–697. © Pleiades Publishing, Ltd., 2016.
Original Russian Text © V.N. Lebedev, 2016, published in Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, 2016, Vol. 56, No. 4, pp. 694–703.

Exponential Examples of Solving Parity Games

V. N. Lebedev
Volgograd State University, Universitetskii pr. 100, Volgograd, 400062 Russia
e-mail: lebedevvn@mail.ru

Received March 14, 2014; in final form, September 29, 2015

Abstract—This paper is devoted to solving certain problems on the computational complexity of deciding the winner in cyclic games. The main result is the proof of the fact that the nondeterministic potential transformation algorithm designed for solving parity games is exponential in terms of computation time.

Keywords: cyclic game, potential transformations, computational complexity, deciding the winner in cyclic games.

DOI: 10.1134/S0965542516040114

1. INTRODUCTION

The main subject of this paper is zero-sum games of complete information, which will be called cyclic. Let a directed graph without deadlock vertices

$$(V : A \cup B; E; h' \to [-h, h]; v \in V)$$

be given in which V is the set of vertices, E is the set of directed edges, $h' : E \to [-h, h], \{-h, -h+1, \ldots, h\}$ is an integer weighting function on the edges, and $v \in V$ is the initial vertex of the game. The absence of deadlocks means that each vertex has at least one outgoing edge incident upon it. The vertices are divided into two nonoverlapping sets—the vertices in which the first player A makes a move (white vertices), and the vertices in which the second player B makes a move (black vertices). A token is moved by the players from the initial vertex $v \in V$ through the graph edges. The game continues until a cycle occurs (i.e., as soon as the token comes to a vertex where it has already been, the game is over). In this paper, we mainly consider three classes of cyclic games. In sign of weight cyclic games, the first player wins if the total weight of the cycle edges is nonnegative; otherwise, the second player wins. In the parity cyclic game, the first player wins if the maximum weight of the cycle edges is even; otherwise, the first player loses. In the mean payoff cyclic game, the payoff of the first player is the mean weight of the edges in the cycle.

In [1], a potential transformation algorithm was proposed to solve the mean cost cyclic games.

First, an upper exponential bound on the computation time depending on the number of vertices was obtained. Second, computational experiments showed that the average number of elementary iterations of the algorithm does not exceed the number of vertices in the graph more than by several fold. Third, examples of problems in which the number of iterations of the algorithm is exponential in the problem size are found.

In [2], a pseudopolynomial bound on the complexity of a more general potential transformation algorithm was found.

Another known fact is that these problems are in the class $NP \cap co-NP$ (see [3]) (parity games are included in the narrower class $UP \cap co-UP$ (see [4]).

Parity games can be reduced to a sign of weight cyclic game in polynomial time.

Efficiency considerations are discussed in [5], where the deterministic potential transformation algorithm is considered. To reduce the algorithm execution time, it is proposed to use the rich group of equivalent potential transformations. First, a nondeterministic potential transformation is employed, and the game is reduced to the canonical form by a deterministic algorithm.

The potential transformations preserve cycle lengths; therefore, we obtain a set of problems that are equivalent to the original problem. A natural conjecture that the average time of solving the problems in this set is polynomial occurs. This conjecture was confirmed by computational experiments with known hard examples (see [5]). It was proposed to use the nondeterministic algorithm for solving parity games. The main issue considered in [5] is whether the nondeterministic Gurvich–Karzanov–Khachiyan's algorithm is polynomial for parity games. In the present paper, a negative answer to this question is given. The justification is as follows.

First, it is shown that the exponential increase in weight magnitudes repeats on subnetworks in the course of the deterministic algorithm execution. This helps prove that the deterministic algorithm has an exponential execution time on a sequence of problems with the exponential growth of weights in networks with a fixed graph structure. (The intricate examples in [1] are a specific case of the exponential growth of the network weight magnitudes).

Second, it is shown that the randomness is lost, i.e., beginning with an arbitrary equivalent network, the same game network is obtained in the course of the algorithm execution with a high probability. Therefore, it is highly probable that the property of exponential weight growth repeats on subnetworks of the original problem. This fact enables us to construct examples in which the execution time of the nondeterministic algorithm is almost surely exponential in the size of the original problem.

Third, we use the fact that the sign of weight cyclic games to which the parity games can be reduced in polynomial time necessarily have an exponential weight growth. Therefore, it is easy to construct a hard exponential sequence of parity games for the nondeterministic potential transformation algorithm. Precise formulations of these claims are given below.

2. STATEMENT OF THE PROBLEM

Let the game network $(V : A \cup B; E; h' \to [-h, h]; v \in V)$ be given. The set of directed edges of the graph of the game that begins in the set of vertices $W \subseteq V$ and ends in the set of vertices $H \subseteq V$ will be denoted by $E(W, H)$. The directed edge with the head at the vertex $w \in V$ and the tail at the vertex $v \in V$ is denoted by $(w, v) \in E$, and its weight is denoted by $[w, v] \in Z$. The subgraph of the game graph $G = (V : A \cup B; E)$ with the subset of vertices $W \subseteq V$ and the subset of edges $I \subseteq E, I \subseteq W \times W$, is denoted by $(W; I)$. The subgraph of G determined by the set of vertices $W \subseteq V$ is denoted by $G(W)$—this is a subgraph $(W \subseteq V; E(W, W))$.

The outgoing neighborhood of the vertex $v \in V$ is denoted by $\mathrm{out}(v) \subseteq V\{w : (v, w) \in E\}$:

$$\mathrm{ext}(v) = \max\{[v, w] : w \in \mathrm{out}(v),\ v \in A\}, \quad \mathrm{ext}(v) = \min\{[v, w] : w \in \mathrm{out}(v),\ v \in B\},$$

$$\mathrm{VEXT}(v) = \{w : w \in \mathrm{out}(v),\ [vw] = \mathrm{ext}(v)\}.$$

In what follows, we will consider sign of weight cyclic games and the use of potential transformations for such games. A definition of the potential transformation algorithm can be found in [1]. The aim of the algorithm is to level local extrema in the vertices using potential transformations $\varepsilon : V \to Q; [v, w]' = [v, w] + \varepsilon(v) - \varepsilon(w)$. Then, the locally optimal strategies in the transformed game are globally optimal. The potential transformations preserve cycle lengths; therefore, the found strategies are also optimal in the original game. To understand the results of this paper, the reader must know the algorithm described in [1]. Let us give a brief description of this algorithm. We will consider the auxiliary algorithm presented in [1] in the context of parity games. The parity games can be reduced to sign of weight cyclic games; for this reason, we assume that the boundary is fixed and equal to zero as in [5]. The main points of the construct are as follows.

The auxiliary algorithm is iterative. The vertices are divided into two nonoverlapping classes L and $V - L = \overline{L}$. The first class L is called the *labeled set of vertices*. It consists of the vertices with a negative local extremum and the vertices of critical (zero) local extrema from which the second player can forcedly (i.e., independently of how the opponent plays) push the game to the vertices with negative extrema. The complementary class $\overline{L}$ is defined similarly relative to the second player. The brief formal definition is $L_0 = V^- = \{v \in V : \mathrm{ext}(v) < 0\}$. Next, we compose the sets $L_1, \dots, L_i$ from the critical vertices $V^0 = \{v \in V : \mathrm{ext}(v) = 0\}$ using the following inductive rule.

Let the sets $L_0, \dots, L_i$, $H_i = L_0 \cup, \dots, \cup L_i$, $i \ge 0$, have already be constructed. Then,

$$L_{i+1} = \{v \in \overline{H_i} \cap B : VEXT(v) \cap L_i \neq 0\} \cup \{v \in \overline{H_i} \cap A : VEXT(v) \subseteq L_i\}, \quad i = 1, \dots .$$

The complementary class $\overline{L}$ is represented similarly relative to the first player. Next, for the partition thus obtained, a potential shift under which the set of critical vertices does not decrease (increases at best) is determined. For each white vertex, $\varepsilon(v) = -\max\{c(v,w)v \in A \cap L,\ w \in \overline{L}\}$ is a feasible potential. If there are no boundary edges$\{(v,w)v \in L, w \in \overline{L}\} = 0$, then the feasible potential in such a white vertex is assumed to be infinite.

Consider the black vertices V^-. Let v be the current black vertex such that, if there is at least one edge with a nonpositive weight leading to the set L incident upon this vertex, then its potential is infinite. Otherwise, the feasible potential of this vertex is its local extremum taken with the negative sign. The feasible potential of any critical black vertex is infinite. The feasible potentials of the vertices in the complementary set are determined similarly. Next, the minimum ε of these potentials is found, the potential transformation of the edge weights $[v,w]' = [v,w] + \varepsilon,\ v \in L,\ w \in \overline{L}$ and $[w,v]' = [w,v] - \varepsilon,\ v \in L,\ w \in \overline{L}$, is performed, and this completes the iteration step. If the minimum potential is infinite or if one of the sets V^-, V^+ is empty, then the auxiliary algorithm terminates. The basic finiteness assertion is that the execution time of the auxiliary algorithm does not exceed $|V|^{|V|}$ iteration steps.

Remark 1. The feasible potentials of the white vertices in the labeled set L are determined in the same way independently of whether these vertices are critical or insufficient V^-. The feasible potentials of the black vertices in the complement $\overline{L}$ are determined in the same way independently of whether these vertices are critical or redundant V^+. The nondeterministic potential transformation algorithm first equiprobably assigns integer numbers $\varepsilon(v) \in [-h,h]$ to all the vertices $v \in V$ of the game network, then applies the potential transformation according to these ε, and then applies the ordinary potential transformation algorithm [1] with the zero boundary of critical vertices. Since potential transformations preserve the cycle lengths, the validity of the nondeterministic algorithm is obvious.

Definition 1. The *game network* G_{n+1} is defined as the triple ($V = \{a_1,\ldots,a_n; b_1,\ldots,b_n\}$; $E = \{(a_i b_{i+1}),\ldots,(a_i,b_n),\quad (b_i a_{i+1}),\ldots,(b_i,a_n),\quad i = 1,\ldots,n-1;\quad (a_n,b_{n+1}),(b_n,a_{n+1});\quad (a_{n+1},a_{n+1}),(b_{n+1},b_{n+1})$; $2\min\{[\![a_i b_{i+1}]\!],[b_i a_{i+1}]\} \leqslant \max\{[\![a_i b_{i+1}]\!],[b_i a_{i+1}]\},\quad i = 1,\ldots,n;\quad 2\max\{[\![a_i b_{i+1}]\!],[b_i a_{i+1}]\} \leqslant \min\{[\![a_{i+1} b_{i+2}]\!],[b_{i+1} a_{i+2}]\},\quad i = 1,\ldots,n-1,\quad [a_1 b_i] = [a_2 b_i] = \ldots = [a_{i-1} b_i] < 0,\quad i = 2,\ldots,n,\quad [a_n,b_{n+1}] < 0$, $[a_{n+1},a_{n+1}] = -1$, $[b_1 a_i] = [b_2 a_i] = \ldots = [b_{i-1} a_i] < 0$, $i = 2,\ldots,n$), $[b_n,a_{n+1}] > 0$, $[b_{n+1},b_{n+1}] = +1$.

The game network G_{n+1} is an almost acyclic graph. All the edges lead to vertices with greater indexes, except for the last white and black vertices. The edges incident upon the last white and black vertices form two loops. The incoming edges of the last vertices begin only in the next to last vertices. The edges that lead to the white vertex from all the preceding black vertices have the same positive weight, and the edges that lead to the black vertex from all the preceding white vertices have the same negative weight. The magnitudes of the edge weights increase not slower than the geometric sequence with the quotient 2.

Definition 2. The *game network* H_n is defined as the quadruple ($V = \{a_1,\ldots a_n; b_1,\ldots,b_n\}$; $E = \{(a_i b_{i+1}),\ldots,(a_i,b_n),\quad (b_i a_{i+1}),\ldots,(b_i,a_n),\quad i = 1,\ldots,n-1;\quad (a_n,b_n),(b_n,a_n);\quad 2\min\{[\![a_i b_{i+1}]\!],[b_i a_{i+1}]\} \leqslant \max\{[\![a_i b_{i+1}]\!],[b_i a_{i+1}]\}$; $2\max\{[\![a_i b_{i+1}]\!],[b_i a_{i+1}]\} \leqslant \min\{|[a_{i+1} b_{i+2}]|,[b_{i+1} a_{i+2}]\}$, $i = 1,\ldots,n-2$, $2\max\{[\![a_{n-1} b_n]\!]$, $[b_{n-1} a_n]\} \le -[a_n b_n] = [b_n a_n], [a_1 b_i] = [a_2 b_i] = \ldots = [a_{i-1} b_i] < 0$, $i = 2,\ldots,n$, $[b_1 a_i] = [b_2 a_i] = \ldots = [b_{i-1} a_i] > 0$, $i = 2,\ldots,n$).

The game network H_n is an almost acyclic graph. All the edges lead to vertices with greater indexes, except for the last white and black vertices. The edges incident upon the last white and black vertices form a unique cycle. The edges that lead to a white vertex from all the preceding black vertices have the same positive weight, and the edges that lead to the black vertex from all the preceding white vertices have the same negative weight. The magnitudes of the edge weights increase not slower than the geometric sequence with the quotient 2. Figure 1 shows a schematic of the network H_n. To keep the weights not very large, we retained only the vertex where the corresponding edge ends. The minus symbol marks the negative edge weight.

Proposition 1. *There exists a sequence of game networks* $S_n, n \ge 1, |V_n| = poly(n)$, *such that the following conditions are met. Beginning with a positive integer* N, *for the game network* S_n $(n \ge N)$ *with equiprobable*

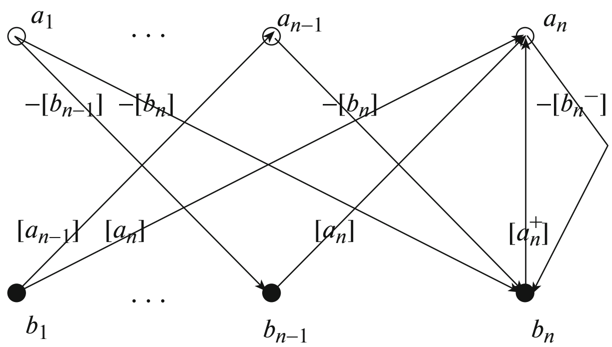

Fig. 1.

potentials $\varepsilon : V \to \{-h, -h+1, \ldots, h-1, h\}$, *for arbitrary nonnegative integer, the number of iterations of the nondeterministic potential transformation algorithm is* $\Omega(2^n)$ *with the probability* $\Omega(1 - 1/\exp(n))$.

The proof immediately follows from Propositions 2 and 3.

Proposition 2. *The deterministic potential transformation algorithm makes* $2^n - 1$ *iterations on the game network* H_n $(n = 2, \ldots)$.

Proposition 3. *Beginning with a positive integer* N, *for any* $n \geq N$, *the nondeterministic potential transformation algorithm makes on the game network* G_{n+1} $\Omega(2^n)$ *iteration with the probability* $\Omega(1/n^4)$.

Proof of Proposition 3. For definiteness, let $\max\{-[a_n b_{n+1}], [b_n a_{n+1}]\} = [b_n a_{n+1}]$.

Consider favorable outcomes of the distribution of the initial random potentials that favor the long execution time of the algorithm:

(1) $\max\{v \in V : |\varepsilon(v)|\} = -\varepsilon(a_{n+1}) > 0$,

(2) $\max\{v \in V - a_{n+1} : |\varepsilon(v)|\} = \varepsilon(b_{n+1}) > 0$,

(3) $\max\{v \in V - a_{n+1} - b_{n+1} : |\varepsilon(v)|\} = \varepsilon(b_n) > 0$,

(4) $\max\{v \in V - a_{n+1} - b_{n+1} - b_n : |\varepsilon(v)|\} = -\varepsilon(a_n) > 0$.

Consider how the network G_{n+1} is processed.

Proposition 4. *Let j be the iteration step at which the vertex* a_n *becomes critical for the first time. Then, all the vertices except for* b_n, b_{n+1}, a_{n+1} *become critical after this step.*

Proof of Proposition 4. The vertices a_{n+1} and b_{n+1} are always insufficient and redundant, respectively, due to the loop edges incident upon them. Due to the maximum conditions, a_n becomes critical earlier than b_n (the weight $[a_n b_{n+1}]$ is closer to zero than the weight $[b_n a_{n+1}]$).

1. It is clear that at the end of the iteration j the set V^- of insufficient vertices does not contain white vertices, except for a_{n+1}. Let a_i $(i < n)$ be a white vertex that turned out to be in V^- as a result of the initial random transformation. Due to the maximum condition, it holds that $[a_n b_{n+1}] < [a_i b_n] < 0$.

At the initial time, we have, due to the maximum conditions, $b_n \in V^+$, $a_{n+1} \in V^-$, $a_n \in V^-$, $a_i \in V^-$. Therefore, the weights of the edges $-[b_n a_{n+1}]$, $[a_n b_{n+1}]$, $[a_i b_n] < 0$ will increase identically at each iteration. For this reason, the first labeled edge that becomes equal to zero is $[a_i b_n]$ or the vertex a_i will become critical even earlier.

2. Similarly, at the start of the iteration j, the set V^+ does not include black vertices except for b_n, b_{n+1}, $[b_n a_{n+1}] > -[a_n b_{n+1}] > [b_i a_n] > 0$. The labeled edges will decrease. Therefore, the first edge that becomes equal to zero is $[b_i a_n]$ or the vertex b_i will become critical even earlier.

3. We show that, at the start of the iteration j, the set V^- does not contain vertices b_i with $i < n$, and the set V^+ does not contain vertices a_i with $i < n$.

Assume the converse. Let the first vertex in V^- in reversed order $n-1, n-2, \ldots, i$ be b_i. We may assume that V^+ does not contain white vertices with the indices $n-1$, $n-2, \ldots, i+1$ (elsewise, we could reason symmetrically relative to the first vertex in V^+).

The negative edges outgoing the vertex b_i at the start of iteration j can lead only to the complement $\overline{L}$ of the labeled set. If there existed a negative edge leading from b_i to the labeled set L, then it would be possible to descend to the set V^- on extremal zero edges, i.e. to the vertex a_n (see the structure of the labeled set in the description of the algorithm). However, there are no negative paths from b_i to a_n. Indeed, the initial paths are positive (the weight of the last edge in such a path is positive, $[b'a_n] > 0$, and the edge weights increase not slower than a geometric sequence with the quotient 2; therefore, the sign of the total path weight is determined by the sign of the maximum (in the absolute value) edge weight, i.e., $[b'a_n] > 0$). The random addition to the path weight $\varepsilon(b_i) - \varepsilon(a_n) \geq 0$, $i < n$, is nonnegative. In the course of the algorithm execution, the vertices b_i and a_n are in the same labeled set L, therefore, the lengths of the paths from b_i to a_n will remain unchanged (the boundary of L is crossed an even number of times).

Therefore, any negative edge (b_i, a_s) leads to $\overline{L}$. Next, one can reach b_n on zero edges. However, the weight of the resulting path $b_i \to a_s \to b_n$ is closer to zero than the weight of the edge (a_n, b_{n+1}). This is true for the initial graph (due to the exponential growth of the network edge weights). In the course of the algorithm execution, the weight of this path and the weight of the edge (a_n, b_{n+1}) increased by the same quantity. Therefore, the negative weight of the edge (b_i, a_s) is closer to zero than the weight of (a_n, b_{n+1}). Hence, the vertex a_n cannot become critical at iteration j (see the choice of the shift potential in the algorithm description). This completes the proof of Proposition 4.

Proposition 5. *At iteration j, the labeled set L does not contain white vertices (except for a_n and a_{n+1}), and the complement $\overline{L}$ does not contain black vertices (except for b_n and b_{n+1}).*

Proof of Proposition 5. Assume the converse. The sets L, $\overline{L}$ are organized as shown in Fig. 2a.

1. We prove that L does not contain additional white vertices. Assume that L contains an additional white vertex a_i as shown in Fig. 2a. Let us show that the weight $[a_i, b_n]$ is closer to zero than the weight $[a_n, b_{n+1}]$. The solid lines in Fig. 2a show the extremal zero edges:

$$[a_i, b_n]' = [a_i, b_n] + \varepsilon(a_i) - \varepsilon(b_n), \tag{1}$$

where $\varepsilon(a_i)$ and $\varepsilon(b_n)$ are the complete potentials of the vertices a_i and b_n (the initial random ones and those obtained in the course of the algorithm execution):

$$[a_n, b_{n+1}]' = [a_n, b_{n+1}] + \varepsilon(a_n) - \varepsilon(b_{n+1}). \tag{2}$$

Let the path $a_i \ldots b_s a_n$ got a zero weight; then,

$$[b_s, a_n] + \Delta + \varepsilon(a_i) - \varepsilon(a_n) = 0,$$

where Δ is the negative initial weight of the path $a_i \to b_s$. This formula implies $\varepsilon(a_i) = -[b_s, a_n] - \Delta + \varepsilon(a_n)$. Plug this expression into (1) and subtract (2) to obtain

$$[a_i, b_n]' - [a_n, b_{n+1}]' = [a_i, b_n] - [b_s, a_n] - \Delta - \varepsilon(b_n) - [a_n, b_{n+1}] + \varepsilon(b_{n+1}) > 0.$$

Indeed, $[a_i, b_n] - [b_s, a_n] - [a_n, b_{n+1}] > 0$ (due to the exponential growth), $-\Delta > 0$, and $-\varepsilon(b_n) + \varepsilon(b_{n+1}) \geq 0$. The last inequality holds because the vertices b_n and b_{n+1} belonged to $\overline{L}$; therefore, the potential is zero in the course of the algorithm execution; hence, only the initial random potential remains. Since $\varepsilon(b_n)$, $\varepsilon(b_{n+1}) > 0$ are, respectively, the third and the second maximums, we obtain the desired inequality. Since the weights $[a_i, b_n]'$ and $[a_n, b_{n+1}]'$ are negative, we conclude that the weight $[a_i, b_n]'$ is closer to zero than the weight $[a_n, b_{n+1}]'$. The transition of the vertex a_n to the set of critical vertices at iteration j is impossible.

2. Let us show that the set $\overline{L}$ does not contain additional black vertices at the iteration j.

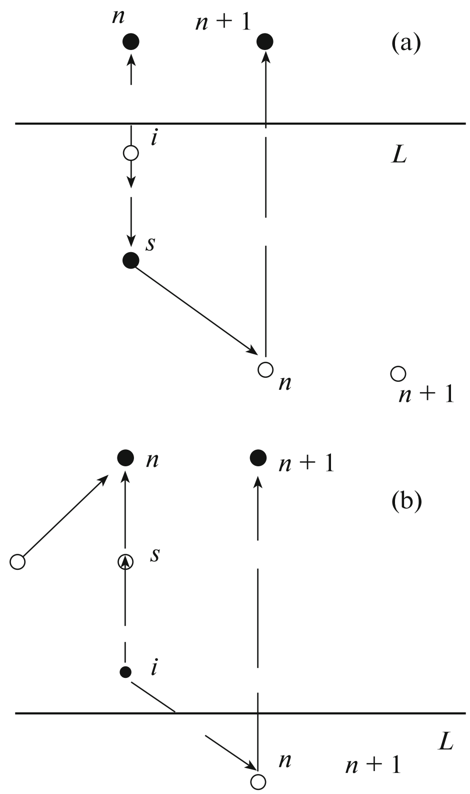

Fig. 2.

Assume that $\overline{L}$ contains an additional black vertex as shown in Fig. 2b. We show that the positive weight $[b_i a_n]'$ is closer to zero than the negative weight $[a_n b_{n+1}]'$:

$$[a_n, b_{n+1}]' = [a_n, b_{n+1}] + \varepsilon(a_n) - \varepsilon(b_{n+1}), \quad (3)$$

$$[b_i, a_n]' = [b_i, a_n] + \varepsilon(b_i) - \varepsilon(a_n). \quad (4)$$

Let the path $b_i \to a_s b_n$ got a zero weight. Then,

$$[a_s, b_n] + \Delta + \varepsilon(b_i) - \varepsilon(b_n) = 0,$$

where Δ is the initial positive weight of the path $b_i \to a_s$. This formula implies $\varepsilon(b_i) = -[a_s, b_n] - \Delta + \varepsilon(b_n)$. Plug this expression into (4) and add (3) to obtain

$$\begin{aligned}[b_i, a_n]' + [a_n, b_{n+1}]' &= [b_i, a_n] + \varepsilon(b_i) - \varepsilon(a_n) + [a_n, b_{n+1}] + \varepsilon(a_n) - \varepsilon(b_{n+1}) \\ &= [b_i, a_n] - [a_s, b_n] - \Delta + \varepsilon(b_n) + [a_n, b_{n+1}] - \varepsilon(b_{n+1}) < 0.\end{aligned}$$

Indeed, $[a_s, b_n] - [b_i, a_n] - [a_n, b_{n+1}] > 0$ (due to the exponential growth); $-\Delta < 0$; $-\varepsilon(b_n) + \varepsilon(b_{n+1}) \geqslant 0$. The latter inequality holds because the vertices b_n and b_{n+1} were in the set $\overline{L}$; therefore, the potential during the algorithm operation is zero, and only the initial random potential remains. Since $\varepsilon(b_n)$, $\varepsilon(b_{n+1}) > 0$ are, respectively, the third and the second maximums, we obtain the desired inequality. If the weights $[a_i, b_n]'$ and $[a_n, b_{n+1}]'$ are negative, then we conclude that the weight $[a_i, b_n]'$ is closer to zero than $[a_n, b_{n+1}]'$. Hence, the transition of the vertex a_n to the set of critical vertices at iteration j is impossible.

Therefore, the positive weight $[b_i a_n]$ is closer to zero than the negative weight $[a_n b_{n+1}]$. This completes the proof of Proposition 5.

Proof of Proposition 3. After the next iteration step $j + 1$, there emerges the zero transition from the black vertex b_n to the vertex a_{n+1}. Therefore, we obtain the following organization of the labeled set and its complement. The labeled set has the following form: the white vertices $a_1, \ldots, a_{n-1}$ have zero transitions to the black vertex b_n, which, in turn, has a zero transition to the vertex a_{n+1}, and there is a negative loop incident upon the latter vertex. The complement of the labeled set has a symmetric form: the black vertices $b_1, \dots, b_{n-1}$ have zero transitions to the white vertex a_n, which, in turn, has a zero transition to the vertex b_{n+1}, and there is a positive loop incident upon the latter vertex.

Since the weights in the initial graph $[a_1, b_n], \dots, [a_{n-1}, b_n]$ are identical and the weights $[b_1, a_n], \dots, [b_{n-1}, a_n]$ are also identical, the common potentials of the vertices $a_1, \dots, a_{n-1}$ are identical and the potentials of the vertices $b_1, \dots, b_{n-1}$ are identical as well. Therefore, the subnetwork determined by the vertices $a_1, \dots, a_{n-1}, b_1, \dots, b_{n-1}$ repeats the game network H_{n-1} up to the initial shift $\varepsilon(a_1) =, \dots, = \varepsilon(a_{n-1}) < 0$; $\varepsilon(b_1) =, \dots, = \varepsilon(b_{n-1}) = 0$. The vertices a_{n-1}, b_n, and a_{n+1} always remain in L, and the vertices b_{n-1}, a_n, and b_{n+1} remain in $\bar{L}$. By Proposition 2, the number of the remaining iteration steps on such a network of the potential transformation algorithm is $\Omega(2^n)$.

The asymptotic probability of favorable outcomes can be easily obtained using the integral estimation of the corresponding sums. Thus, Proposition 3 is proved.

Proof of Proposition 2. The proof is by induction on the parameter n. For $n = 2$, the potential transformation algorithm makes three iteration steps. Suppose that, for the game network H_n, the algorithm makes $2^n - 1$ iteration steps, where $n \geqslant 2$. Let us prove that this also holds for $n + 1$. Assume that $-[a_n, b_{n+1}] < [b_n, a_{n+1}]$ and $[a_1, b_2]$ is the closest to zero weight of edges in H_{n+1}.

Consider the first iteration step j at which the vertex a_n first becomes critical. It is even simpler to prove that all the vertices, except for a_{n+1}, b_n and b_{n+1}, are critical after iteration j. Proposition 4 is proved using a minor modification. When the vertex a_n got a zero weight edge leading to b_{n+1} for the first time, there are no other vertices with extremal zero weight edges that lead to the vertices b_{n+1} and a_{n+1} (if we conversely assume that there is a vertex a_i, $i < n$, $[a_i, b_{n+1}] = 0$ at iteration j, then $[a_i, b_n] > 0$ because the edges $[a_i, b_n]$, $[a_i, b_{n+1}] = 0$ were changed identically; in the same way, there exists no vertex b_i, $i < n$, $[b_i, a_{n+1}] = 0$ at iteration j). Next, we exactly repeat the proof of Proposition 4.

By the inductive assumption, $j = 2^n - 1$. Indeed, at the iterations $1, \dots, j - 1$, the vertices b_n and b_{n+1} are redundant and the vertices a_n and a_{n+1} are insufficient; therefore, the weights $[v_i v_n]$, $[v_i v_{n+1}]$ for $i = 1, \dots, n - 1$ and $[v_n v_{n+1}]$ change identically. Therefore, the additional edges $(v_i v_{n+1})$ do not affect the processing of the subnetwork consisting of the first n vertices $a_1, \dots, a_n b_1, \dots, b_n$ at the first j iterations.

After the next iteration $j + 1$, a zero transition from the black vertex b_n to the vertex a_{n+1} emerges. Therefore, the labeled set and its complement are organized as follows. The labeled set consists of the white vertices $a_1, \dots, a_{n-1}$, which have zero transitions to the black vertex b_n, which, in turn, has a zero transition to a_{n+1}.

The complement of the labeled set has a symmetric form: the black vertices $b_1, \dots, b_{n-1}$ have zero transitions to the white vertex a_n, which, in turn, has a zero transition to the vertex b_{n+1}. Now we prove that the subnetwork defined by the vertices $a_1, \dots, a_{n-1}, a_{n+1} b_1, \dots, b_{n-1} b_{n+1}$ is an H_n network on n vertices up to the initial shift.

Find the weighting function of the edges of this subnetwork H_n. The potentials can be reconstructed from the extremal zero edges. Various orders of the absolute weights of edges are considered in a similar fashion.

Consider the case when $[b_n, a_{n+1}] \geqslant -2[a_n, b_{n+1}]$ and $[a_1, b_2]$ is the closest to zero weight of the edges of the network H_{n+1}.

Consider the potentials of the following vertices:

$\varepsilon(b_{n+1}) = 0$—this vertex is always in the complement $\bar{L}$ of the labeled set;

$\varepsilon(a_n) = -[a_n, b_{n+1}]$—the edge's weight is set to zero;

$\varepsilon(b_n) = 0$—this vertex is always in the complement $\bar{L}$ of the labeled set;

$\varepsilon(a_{n+1}) = [b_n, a_{n+1}]$—the weight of the incoming edge (b_n, a_{n+1}) is set to zero;

$\varepsilon(b_1) = \dots = \varepsilon(b_{n-1}) = -[b_{n-1}, a_n] - [a_n, b_{n+1}]$—the weights of the edges (b_i, a_n) for $i = 1, \dots, n - 1$ are set to zero;

$\varepsilon(a_1) = \dots = \varepsilon(a_{n-1}) = -[a_{n-1}, b_n]$—the weights of the edges (a_i, b_n) for $i = 1, \dots, n - 1$ are set to zero.

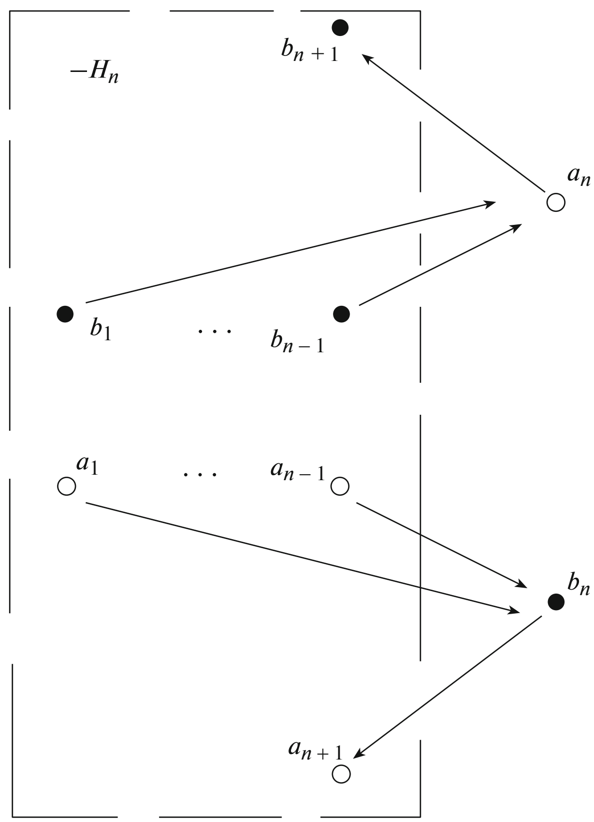

Fig. 3.

At the iteration $j + 1$, which we now consider, the set of labeled vertices L and its complement $\overline{L}$ have the form $a_1, \ldots, a_{n-1}, b_n, a_{n+1}$ and $b_1, \ldots, b_{n-1}, a_n, b_{n+1}$, respectively (Fig. 3). In the resulting network, the weight of the boundary edge that is closest to zero is

$$[a_1, b_2]' = [a_1, b_2] - [a_{n-1}, b_n] + [b_{n-1}, a_n] + [a_n, b_{n+1}].$$

To prove this fact, we make the shift

$$\varepsilon(L) = -[a_1, b_2]' + [a_1, b_2] = [a_{n-1}, b_n] - [b_{n-1}, a_n] - [a_n, b_{n+1}];$$

then, we obtain partitions of the set of labeled vertices L and its complement $\overline{L}$ ($L = \{a_1, \ldots, a_{n-1}, b_n, a_{n+1}\}$ and $\overline{L} = \{b_1, \ldots, b_{n-1}, a_n, b_{n+1}\}$).

Below we show that, in the network determined by the vertices $\{a_1, \ldots, a_{n-1}, a_{n+1}\}$ and $\{b_1, \ldots, b_{n-1}, b_{n+1}\}$, the weights grow exponentially as required (the weight $[a_1, b_2]$ is the closest to zero). Therefore, at the iteration $j + 1$, the closest to zero weight in the network is $[a_1, b_2]'$. By the inductive assumption, the remaining execution time of the algorithm is $2^n - 1$ iterations on the network H_n. (The black vertex n will be labeled and the white vertex n will be unlabeled in the course of the remaining algorithm operation because there always is an extremal zero transition to V^- and V^+, respectively. Therefore, the subnetwork determined by the vertices $\{a_1, \ldots, a_{n-1}, a_{n+1}\}$ and $\{b_1, \ldots, b_{n-1}, b_{n+1}\}$ is processed in the same way as H_n. This can be proved by induction on the iteration index using the remark on the description of the potential transformation algorithm.) Let us find the new weights of the edges:

$$[a_{n+1}, b_{n+1}]' = [a_{n+1}, b_{n+1}] + [b_n, a_{n+1}] + [a_{n-1}, b_n] - [b_{n-1}, a_n] - [a_n, b_{n+1}],$$

$$[b_{n+1}, a_{n+1}]' = [b_{n+1}, a_{n+1}] - [b_n, a_{n+1}] - [a_{n-1}, b_n] + [b_{n-1}, a_n] + [a_n, b_{n+1}],$$

$$[a_{n-1}, b_{n+1}]' = [a_{n-1}, b_{n+1}] - [a_{n-1}, b_n] + [a_{n-1}, b_n] - [b_{n-1}, a_n] - [a_n, b_{n+1}] = -[b_{n-1}, a_n],$$

$$[b_{n-1}, a_{n+1}]' = [b_{n-1}, a_{n+1}] - [b_{n-1}, a_n] - [a_n, b_{n+1}] - [b_n, a_{n+1}] - {}$$
$$- [a_{n-1}, b_n] + [b_{n-1}, a_n] + [a_n, b_{n+1}] = -[a_{n-1}, b_n],$$

$$[b_i, a_j]' = [b_i, a_j], \quad i = 1, \dots, n-2, \quad j = i+1, \dots, n-1,$$

$$[a_i, b_j]' = [a_i, b_j], \quad i = 1, \dots, n-2, \quad j = i+1, \dots, n-1.$$

To verify the exponential growth, we use the following fact. The sign of any subset (without repetition of elements) of the set of integers with an exponential growth of their absolute values (the quotient of the geometric sequence is not less than 2) is defined as the sign of the maximum absolute value of the numbers in this subset. Therefore, the total execution time of the algorithm is $2^n + 2^n - 1 = 2^{n+1} - 1$. Proposition 2 is thus proved.

The basic Proposition 1 is derived as follows. Create n^5 copies of the network G_n. This is the game network S_n. The execution time of the algorithm on a set of disjoint networks is not less than its execution time on each of its components. It is highly probable that at least on one of these subnetworks the distribution of the maximum values of the potentials will be favorable. The probability of failure is estimated in the standard way by

$$e^{n^5 \ln(1-\text{const}/n^4)} \le e^{n^5(-\text{const}/n^4)} \le e^{-\text{const}(n)}, \quad \text{const} > 0.$$

This completes the proof of Proposition 1.

Here is the formulation of the main lemma.

Lemma 1. *Let S_n' be a sequence of game networks with the same structure of graphs as in the sequence in Proposition* 1. *The weighting function in each copy is the same. The black vertices* 1, 2, ..., $n + 1$ *have the odd weights* 1, 3, ..., *respectively; and the white vertices* 1, 2, ..., $n + 1$ have the even weights 2, 4, ... *The nondeterministic potential transformation algorithm applied to the problem obtained by the natural reduction using a transformation that preserves the signs of subsets has an exponential execution time on this sequence of problems with the unit asymptotic probability.*

Proof. For the nth equivalent cyclic game, the growth of the absolute values of the weights is not lower than that of the geometric sequence with the quotient $n^5 > 2$, where $n > 1$; i.e., the equivalent game network is a G_n network. By Proposition 1, the execution time of the potential transformation algorithm is $\Omega(2^n)$ with the probability $\Omega(1 - 1/\exp(n))$.

This implies that the nondeterministic potential transformation algorithm with a fixed zero boundary of the critical vertices is exponential with the unit asymptotic probabilityc. The fixed zero boundary can be obtained with probability 1.

Remark 2. In the general statement proposed in [1] when the boundary is set at the half of the sum of the maximum and minimum extrema, the nondeterministic potential transformation algorithm is exponential.

Consider the network S_n, i.e., n^5 copies of game networks G_n (in every G_n, the weights of edges grow exactly as the geometric sequence with the quotient 2, and the maximum weight is 2^n). The number of vertices in S_n is n^k, where $k = \text{const}$. Add to S_n the components $K_1, \dots, K_m$, where $m = 25n^{k+1}n^{k+2}$. Each component is a pair of a white and black vertices connected by a pair of edges. The weight of the edge leading from the white to the black vertex is $+8*2^n$, and the weight of the edge leading from the black vertex to the white one is $-8*2^n$. Thus, we obtain the game network S_n' on which the potential transformation algorithm makes $\Omega(2^n)$ iteration steps with the asymptotic probability 1. Let $N = \text{poly}(n)$ be the number of vertices in the network. After a random distribution of potentials, the zero boundary will be set with the unit asymptotic probability. The addition of components does not accelerate the processing of the copies G_n because the graph is unconnected.

It suffices to show that the maximum and minimum extrema occur in one of the added components due to the large number of these components. Therefore, the maximum and minimum extrema in the entire game network are symmetric about zero. Firstly, we may assume that the bound of the random potentials h is not less than 2^n (otherwise, the maximum and minimum extrema would be attained in the additional components, and the bound in this case would be zero because all the weights of loops in these components are zero).

Secondly, the probability of two identical potentials is asymptotically equal to zero. The probability of the opposite event is

$$h(h-1)\dots(h-N+1)/h^N \xrightarrow{n\to\infty} 1$$

($h(n)$ grows exponentially with n, and $N(n)$ grows polynomially).

Thirdly, we can guarantee that the added components contain a large region with regular potentials (positive potentials in white vertices and negative potential in black vertices) with probability 1. It can be shown that the number of regular components is at least 1/25th of all m components with the asymptotic probability 1; i.e., there are $n^{k+1}n^{k+2}$ regular components.

Fourthly, let us call a set of n^{k+1} regular components an assembly. Then, we have n^{k+2} assemblies of which each contains n^{k+1} regular components with probability 1. The assemblies are large to guarantee that the maximum and minimum potentials in them are greater than in the main part of S_n. Indeed, the probability of the complementary event is bounded above by the quantity $n^k/(n^k + n^{k+1})$, which asymptotically tends to zero. Lastly, we use a large number of assemblies to guarantee that the maximum and minimum potentials of the assembly belong to the same component in it.

The probability of identical potentials is zero. Therefore, in each assembly we have equiprobable combinations of n^{k+1} positive potentials with the same number of negative potentials. Hence, the probability that the maximum potential is not in the same component as the minimum potential is not greater than $(1 - 1/n^{k+1})n^{k+2} \xrightarrow{n\to\infty} 0$.

ACKNOWLEDGMENTS

I am grateful to M.N. Vyalyi for valuable remarks and discussion of the result, and I would like to thank the reviewer.

This work was supported in part by the Russian Foundation for Basic Research (project no. 15-01-04146) and by the Russian Academy of Sciences (project no. 3, Division of Mathematical Sciences).

REFERENCES

1. V. A. Gurvich, A. V. Karzanov, and L. G. Khachiyan, "Cyclic games and an algorithm to find minimax cycle means in directed graphs," Zh. Vychisl. Mat. Mat. Fiz. **28** (5), 85–91 (1988).
2. N. Pisaruk, "Mean cost cyclical games," Math. Oper. Res. **59** (4), 817–828 (1999).
3. A. V. Karzanov and V. N. Lebedev, "Cyclical games with prohibitions," Math. Program. **60**, 277–293 (1993).
4. M. Jurdzinski, "Deciding the winner in parity games is in $UP \cap co - UP$," Inf. Proc. Lett. **68**, 119–124 (1998).
5. E. Beffara E. and S. Vorobyov, "Is randomized Gurvich–Karzanov–Khachiyan's algorithm for parity games polynomial?" Technical report, Department of information Technology Uppsala University, 2001.

Translated by A. Klimontovich