2

Max-Min Control Problems and Solving Zero-Sum Games on Networks

The mathematical tool we develop in this chapter allows us to derive methods and algorithms for solving max-min discrete control problems and to determine optimal stationary strategies of the players in dynamic zero-sum games on networks. We propose polynomial-time algorithms for finding max-min paths on networks and determining optimal strategies of players in antagonistic positional games. These algorithms are applied for studying and solving cyclic games. The computational complexity of the proposed algorithms is analyzed.

2.1 Discrete Control and Finite Antagonistic Dynamic Games

We consider a discrete dynamical system L with a finite set of states $X \subseteq \mathbb{R}^n$. Assume that the dynamical system is controlled by two players and it is described as follows:

$$x(t+1) = g_t(x(t), u^1(t), u^2(t)), \quad t = 0, 1, 2, \dots,$$

where

$$x(0) = x_0$$

is a given starting point of system L and $u^1(t) \in \mathbb{R}^{m_1}$, $u^2(t) \in \mathbb{R}^{m_2}$ represent vectors of control parameters of the players 1 and 2, respectively. For the players feasible sets $U_t^1(x(t))$ and $U_t^2(x(t))$ are given at every moment in time t for an arbitrary state $x(t)$, i.e.

$$u^1(t) \in U_t^1(x(t)), \quad u^2(t) \in U_t^2(x(t)) \quad \text{for} \quad t = 0, 1, 2, \dots \quad \text{and} \quad x(t) \in X.$$

Additionally, we assume that the final state x_f is fixed and the dynamical system should reach x_f at time moment $T(x_f)$ such that $T_1 \le T(x_f) \le T_2$.

For given admissible vectors of control parameters $u^1(t)$, $u^2(t)$ of the players the integral-time cost of the system's passage from starting state x_0 to final state x_f is defined in the following way:

$$F_{x_0 x_f}(u^1(t), u^2(t)) = \sum_{t=0}^{T(x_f)-1} c_t(x(t), g_t(x(t), u^1(t), u^2(t)))$$

if $T_1 \leq T(x_f) \leq T_2$; otherwise we put

$$F_{x_0 x_f}(u^1(t), u^2(t)) = \infty.$$

Here $c_t(x(t), g_t(x(t), u^1(t), u^2(t))) = c_t(x(t), x(t+1))$ expresses the cost of system L to pass from state $x(t)$ to state $x(t+1)$ at the stage $[t, t+1]$.

We consider the antagonistic game of two players, i.e. the first player intends to maximize the integral time cost by a trajectory $x(0)$, $x(1)$, $\dots$, , $x(T(x_f)) = x_f$ while the second one intends to minimize the integral-time cost.

The main results we discuss in this chapter are concerned with the existence of saddle points, i.e. we are seeking for vectors $u^{1*}(t) \in U^1,\ u^{2*}(t) \in U^2$, for which

$$F_{x_0 x_f}(u^1(t), u^{2*}(t)) \leq F_{x_0 x_f}(u^{1*}(t), u^{2*}(t)) \leq F_{x_0 x_f}(u^{1*}(t), u^2(t)),$$
$$\forall u^1(t) \in U^1, \forall u^2(t) \in U^2,$$

where $U^1 = \Pi_{t,x(t)} U_t^1(x(t))$, $U^2 = \Pi_{t,x(t)} U_t^2(x(t))$.

In Section 1.3 we have already formulated the main results related to dynamic antagonistic games (Corollary 1.10 of Theorem 1.9). In order to prove these results we shall use the same approach as in Chapter 1 and consider antagonistic dynamic games on networks for which the alternate players' control condition holds. Based on the time-expanded network method we can reduce this problem to antagonistic positional games on networks. The most important results we discuss in this chapter are related to determining the optimal stationary strategies of the players in the zero-sum dynamic c-game on networks which may contain directed cycles. Additionally, we also will consider antagonistic dynamic games with infinite time horizon.

2.2 Max-Min Control Problem with Infinite Time Horizon

In Chapter 1 we have noted that the concept of antagonistic games can be applied to the optimal control problem with infinite time horizon. This leads to the following max-min control problem with infinite time horizon:

Let a dynamic system L with finite set of states $X \subseteq \mathbb{R}^n$ be given. Assume that the dynamics of system L is controlled by two players and it is described by a system of difference equations in the same way as in the previous section. Here we assume that the final state is not given and the control is made on an infinite interval of time $[0, \infty]$. In the control process the first player has the aim to maximize the integral mean cost

$$\lim_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} c_t(x(t), g_t(x(t), u^1(t), u^2(t)))$$

by a trajectory $x_0 = x(0)$, $x(1)$, $x(2)$, $\dots$, $x(\tau)$, $\dots$, while the second player intends to minimize this integral mean cost.

If for given vectors of control parameters of the players $u^1(t), u^2(t)$, $t = 0, 1, 2, \dots$ there exists the given limit, then we will consider it as the value of the payoff function $F_{x_0}(u^1(t), u^2(t))$; otherwise we put $F_{x_0}(u^1(t), u^2(t)) = \infty$.

An important particular case of the max-min control problem with infinite time horizon represents a cyclic game. Cyclic games correspond to the stationary case of max-min control problems when the set of states X is divided into two subsets $X = X_1 \cup X_2$ $(X_1 \cap X_2 = \emptyset)$ such that the functions $g_t(x(t), u^1(t), u^2(t))$, $t = 0, 1, 2, \dots$, satisfy the following condition:

$$g_t(x(t), u^1(t), u^2(t)) = \begin{cases} g^1(x(t), u^1(t)), & \text{if } \ x(t) \in X_1; \\ g^2(x(t), u^2(t)), & \text{if } \ x(t) \in X_2, \end{cases}$$

where

$$u^1(t) \in U^1(x(t)), \quad t = 0, 1, 2, \dots, \quad \text{for } x(t) \in X_1$$

and

$$u^2(t) \in U^2(x(t)), \quad t = 0, 1, 2, \dots, \quad \text{for } x(t) \in X_2.$$

We show that the mathematical tool for studying max-min control problems described in this chapter allows us to elaborate algorithms for solving cyclic games with constant costs of the system's passage from one state to another.

Taking into account that the dynamical system in the considered problems has a finite set of states we formulate and study our max-min problems on dynamic networks.

2.3 Zero-Sum Games on Networks and a Polynomial Time Algorithm for Max-Min Paths Problems

In the previous chapter we have studied dynamic c-games with positive cost functions on the edges. Therefore, we cannot use those results for zero-sum games.

In the following we study zero-sum games of two players with arbitrary cost functions on the edges and propose polynomial-time algorithms for their solving. The main results related to this problem have been obtained in [55, 56, 57, 60, 61, 77].

2.3.1 Problem Formulation

In this section we study the antagonistic dynamic c-game of two players on a network with arbitrary constant cost functions on the edges. This case of the problem corresponds to the max-min paths problem on networks, which generalizes classical combinatorial problems of the shortest and the longest paths in weighted directed graphs. This max-min paths problem arises as an auxiliary one when searching optimal stationary strategies of players in cyclic games. Additionally, we shall use the considered dynamic c-game for studying and solving the zero-sum control problem from Section 1.1.2. The main results are concerned with the existence of polynomial-time algorithms for determining max-min paths in networks as well as the elaboration of such algorithms.

Let $G = (X, E)$ be a directed graph with vertex set X and edge set E. Assume that G contains a vertex $x_f \in X$ such that it is attainable from each vertex $x \in X$, i.e. x_f is a sink in G. On edge set E it is given a function $c : E \to \mathbb{R}$, which assigns a cost c_e to each edge $e \in E$. Additionally, the vertex set is divided into two disjoint subsets X_A and X_B ($X = X_A \cup X_B$, $X_A \cap X_B = \emptyset$), which we regard as position sets of two players.

On G we consider a game of two players. The game starts at position $x_0 \in X$. If $x_0 \in X_A$, then the move is done by the first player, otherwise it is done by the second one. Move means the passage from position x_0 to neighbor position x_1 through edge $e_1 = (x_0, x_1) \in E$. After that, if $x_1 \in X_A$, then the move is done by the first player, otherwise it is done by the second one and so on. As soon as the final position is reached the game is over. The game can be finite or infinite. If the final position x_f is reached in finite time, then the game is finite. In the case that the final position x_f is not reached, the game is infinite. The first player in this game has the aim to maximize $\sum_i c_{e_i}$ while the second one has the aim to minimize $\sum_i c_{e_i}$.

Strictly, the considered game in normal form can be defined as follows: We identify the strategies s_A and s_B of the players with the maps

$$s_A : x \to y \in X(x) \quad \text{for} \quad x \in X_A;$$
$$s_B : x \to y \in X(x) \quad \text{for} \quad x \in X_B,$$

where $X(x)$ represents the set of extremities of the edges $e = (x, y) \in E$, i.e. $X(x) = \{y \in X \mid e = (x, y) \in E\}$. Since G is a finite graph then the sets of strategies of the players

$$S_A = \{s_A : x \to y \in X(x) \text{ for } x \in X_A\};$$
$$S_B = \{s_B : x \to y \in X(x) \text{ for } x \in X_B\}$$

are finite sets. The payoff function $H_{x_0}(s_A, s_B)$ on $S_A \times S_B$ is defined in the following way:

Let in G be a subgraph $G_s = (X, E_s)$ generated by edges of the form $(x, s_A(x))$ for $x \in X_A$ and $(x, s_B(x))$ for $x \in X_B$. Then either a unique directed path $P_s(x_0, x_f)$ from x_0 to x_f exists in G_s or such a path does not exist in G_s. In the second case, in G_s there exists a unique directed cycle C_s, which can be reached from x_0.

For given s_A and s_B we set

$$H_{x_0}(s_A, s_B) = \sum_{e \in E(P_s(x_0, x_f))} c_e,$$

if in G_s there exists a directed path $P_s(x_0, x_f)$ from x_0 to x_f, where $E(P_s(x_0, x_f))$ is a set of edges of the directed path $P_s(x_0, x_f)$. If in G there is no directed path from x_0 to x_f, then we define $H_{x_0}(s_A, s_B)$ as follows. Let $P'_s(x_0, y_0)$ be a directed path, which connects the vertex x_0 with the cycle C_s and $P'_s(x_0, y_0)$ has no other common vertices with C_s except y_0. Then we put

$$H_{x_0}(s_A, s_B) = \begin{cases} +\infty, & \text{if } \sum_{e \in E(C_s)} c_e > 0; \\ \sum_{e \in E(P'_s(x_0, y_0))} c_e, & \text{if } \sum_{e \in E(C_s)} c_e = 0; \\ -\infty, & \text{if } \sum_{e \in E(C_s)} c_e < 0. \end{cases}$$

This game is related to zero-sum positional games of two players and it is determined by the graph G with sink vertex x_f, partition $X = X_A \cup X_B$, cost function $c : E \to \mathbb{R}$ and starting position x_0. We denote the network, which determines this game, by $(G, X_A, X_B, c, x_0, x_f)$. In case when the dynamic c-game is considered for an arbitrary starting position $x \in X$ we shall use the notation (G, X_A, X_B, c, x_f).

In [60, 61] it is shown that if G does not contain directed cycles, then for every $x \in X$ the following equality holds:

$$v(x) = \max_{s_A \in S_A} \min_{s_B \in S_B} H_x(s_A, s_B) = \min_{s_B \in S_B} \max_{s_A \in S_A} H_x(s_A, s_B), \tag{2.1}$$

which means the existence of optimal strategies of the players in the considered game. Moreover, in [60, 61] it is shown that in G there exists a tree $GT = (X, E^*)$ with sink vertex x_f, which gives the optimal strategies of the players in the game for an arbitrary starting position $x_0 \in X$. The strategies of the players are obtained by fixing

$$s_A^*(x) = y, \quad \text{if} \quad (x, y) \in E^* \quad \text{and} \quad x \in X_A \setminus \{x_f\};$$
$$s_B^*(x) = y, \quad \text{if} \quad (x, y) \in E^* \quad \text{and} \quad x \in X_B \setminus \{x_f\}.$$

In the general case for an arbitrary graph G equality (2.1) may fail to hold. Therefore, we formulate necessary and sufficient conditions for the existence of optimal strategies of the players in this game and propose a polynomial-time algorithm for determining the tree of max-min paths from every $x \in X$ to x_f. Furthermore, we show that our max-min paths problem on the network can be regarded as a zero-value ergodic cyclic game. So, the proposed algorithm can be used for solving such games.

In [55, 56] the formulated game on network $(G, X_A, X_B, c, x_0, x_f)$ is named dynamic c-game. Some preliminary results related to this problem have been obtained in [60, 61]. More general models of positional games on networks with p players have been studied in [5, 58, 59, 67, 69, 70, 71].

The considered max-min paths problem can be used for the zero-sum control problem with alternate players' control (see Corollary 1.10). For $p = 2$ on the basis of construction from Section 1.10 we obtain network $(\overline{G}, Z_1, Z_2, \overline{c}, z_0, z_f)$, where $\overline{G} = (Z, \overline{E})$, $Z = Z_1 \cup Z_2$ $(Z_1 \cap Z_2 = \emptyset)$ and $\overline{c} = c_e^1 = -c_e^2$, $\forall e \in E$. This network determines the max-min paths problem, the solution of which corresponds to the solution of the zero-sum control problem.

2.3.2 An Algorithm for Solving the Problem on Acyclic Networks

The formulated problem for acyclic networks has been studied in [56, 60, 61].

Let $G = (X, E)$ be a finite directed graph without directed cycles and a given sink vertex x_f. The partition $X = X_A \cup X_B$ $(X_A \cap X_B = \emptyset)$ of the vertex set of G is given and the cost function $c : E \to \mathbb{R}$ on the edges is defined. We consider a dynamic c-game on G with a given starting position $x \in X$.

It is easy to observe that for fixed strategies of the players $s_A \in S_A$ and $s_B \in S_B$ the subgraph $G_s = (X, E_s)$ has the structure of a directed tree with sink vertex $x_f \in X$. This means that the value $H_x(s_A, s_B)$ is determined uniquely by the sum of edge costs of the unique directed path $P_s(x, x_f)$ from x to x_f. In [60, 61] it is proved that for an acyclic c-game on network $(G, X_A, X_B, c, x_0, x_f)$ there exist strategies of the players s_A^*, s_B^* such that

$$
\begin{aligned}
v(x) = H_x(s_A^*, s_B^*) &= \max_{s_A \in S_A} \min_{s_B \in S_B} H_x(s_A, s_B) \\
&= \min_{s_B \in S_B} \max_{s_A \in S_A} H_x(s_A, s_B)
\end{aligned}
\tag{2.2}
$$

and s_A^*, s_B^* do not depend on a starting position $x \in X$, i.e. (2.2) holds for every $x \in X$.

The equality (2.2) is evident in the case that $\mathrm{ext}(c, x) = 0$, $\forall x \in X \setminus \{x_f\}$, where

$$
\mathrm{ext}(c, x) = \begin{cases} \max\limits_{y \in X(x)} c_{(x,y)}, & x \in X_A; \\ \min\limits_{y \in X(x)} c_{(x,y)}, & x \in X_B. \end{cases}
$$

In this case $v(x) = 0$, $\forall x \in X$ and the optimal strategies of the players can be obtained by fixing the maps $s_A^* : X_A \setminus \{x_f\} \to X$ and $s_B^* : X_B \setminus \{x_f\} \to X$ such that $s_A^* \in \mathrm{VEXT}(c,x)$ for $x \in X_A \setminus \{x_f\}$ and $s_B^* \in \mathrm{VEXT}(c,x)$ for $x \in X_B \setminus \{x_f\}$, where

$$\mathrm{VEXT}(c,x) = \{y \in X(x) \mid c_{(x,y)} = \mathrm{ext}(c,x)\}.$$

If network $(G, X_A, X_B, c, x_0, x_f)$ has the property that $\mathrm{ext}(c,x) = 0$, $\forall x \in X \setminus \{x_f\}$, then it is named network in canonic form. So, for the acyclic c-game on a network in canonic form equality (2.2) holds and $v(x) = 0$, $\forall x \in X$.

In the general case equality (2.2) can be proved by using the properties of the potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ on edges $e = (x,y)$ of the network, where $\varepsilon : X \to \mathbb{R}$ is an arbitrary real function on X (the potential transformation for positional games has been introduced in [8, 40, 56]). The fact is that such a transformation of costs on the edges of a acyclic network in a c-game does not change the optimal strategies of the players, although the values $v(x)$ of positions $x \in X$ are changed by $v(x)+\varepsilon(x_f)-\varepsilon(x)$. It means that for an arbitrary function $\varepsilon : X \to \mathbb{R}$ the optimal strategies of the players in acyclic c-games on the networks $(G, X_A, X_B, c, x_0, x_f)$ and $(G, X_A, X_B, c', x_0, x_f)$ are the same.

Note that the vertices $x \in X$ of the acyclic graph G can be numbered with $1, 2, \ldots, |X|$, such that if $x > y$ then in G there is no directed path from y to x. Therefore, we can use the following recursive formula:

$$\varepsilon(x_f) = 0;$$

$$\varepsilon(x) = \begin{cases} \max_{y \in X(x)}\{c_{(x,y)} + \varepsilon(y)\} & \text{for } \; x \in X_A \setminus \{x_f\}; \\ \min_{y \in X(x)}\{c_{(x,y)} + \varepsilon(y)\} & \text{for } \; x \in X_B \setminus \{x_f\} \end{cases} \tag{2.3}$$

to tabulate the values $\varepsilon(x)$, $\forall x \in X$. It is evident that the transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ satisfies the condition $\mathrm{ext}(c',x) = 0$, $\forall x \in X$. This means that the following theorem holds:

Theorem 2.1. *For an arbitrary acyclic network $(G, X_A, X_B, c, x_0, x_f)$ with a sink vertex x_f there exists a function $\varepsilon : X \to \mathbb{R}$ which determines the potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ on edges $e = (x,y)$ such that network $(G, X_A, X_B, c, x_0, x_f)$ has canonic form. The values $\varepsilon(x)$, $x \in X$, which determine function $\varepsilon : X \to \mathbb{R}$, can be found by using recursive formula (2.3).*

On the basis of this theorem the following algorithm for determining optimal strategies of the players in a c-game is proposed in [56].

Algorithm 2.2. **Determining the Optimal Strategies of the Players on an Acyclic Network**

1. Find values $\varepsilon(x)$, $x \in X$, according to recursive formula (2.3) and the corresponding potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ on edges $(x, y) \in E$.
2. Fix arbitrary maps $s_A^*(x) \in \mathrm{VEXT}(c', x)$ for $x \in X_A \setminus \{x_f\}$ and $s_B^*(x) \in \mathrm{VEXT}(c', x)$ for $x \in X_B \setminus \{x_f\}$.

Remark 2.3. The values $\varepsilon(x)$, $x \in X$, represent the values of the acyclic c-game on $(G, X_A, X_B, c, x_0, x_f)$ with starting position x, i.e. $\varepsilon(x) = v(x)$, $\forall x \in X$. Algorithm 2.2 needs $O(|X|^2)$ elementary operations, because the tabulation of the values $\varepsilon(x)$, $x \in X$, using formula (2.3) for acyclic networks needs this number of operations.

2.3.3 Main Results for the Problem on an Arbitrary Network

First of all we give an example which shows that equality (2.1) may fail to hold. In Fig. 2.1 it is given a network with starting position $x_0 = 0$ and final position $x_f = 3$, where positions of the first player are represented by circles and positions of the second player are represented by squares; values of cost functions on edges are given alongside them.

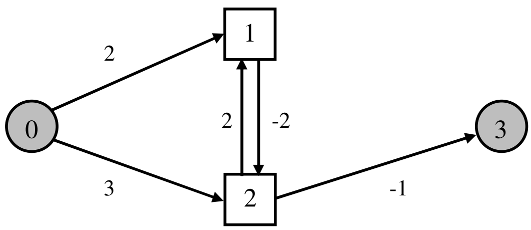

Fig. 2.1.

It is easy to observe that

$$\max_{s_A \in S_A} \min_{s_B \in S_B} H_0(s_A, s_B) = 2, \qquad \min_{s_B \in S_B} \max_{s_A \in S_A} H_0(s_A, s_B) = 3.$$

The following theorem gives conditions for the existence of a saddle point with finite $v(x)$ for each $x \in X$ in the c-game.

Theorem 2.4. *Let $(G, X_A, X_B, c, x_0, x_f)$ be an arbitrary network with sink vertex $x_f \in X$. Additionally, assume that $\sum_{e \in E(C_s)} c_e \neq 0$ for every directed cycle C_s from G. Then for a c-game on $(G, X_A, X_B, c, x_0, x_f)$ condition (2.1)* *with finite* $p(x)$ *holds for every* $x \in X$ *if and only if there exists a function* $\varepsilon : X \to \mathbb{R}$*, which determines a potential transformation* $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ *on edges* $(x, y) \in E$ *such that* $\mathrm{ext}(c', x) = 0$*,* $\forall x \in X$*. Moreover, if in* G *there exists a potential transformation* $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ *on edges* $(x, y) \in E$ *such that* $\mathrm{ext}(c', x) = 0$*,* $\forall x \in X \setminus \{x_f\}$*, then* $v(x) = \varepsilon(x) - \varepsilon(x_f)$*,* $\forall x \in X$*.*

Proof. $\Longrightarrow$ Let us consider that $\sum_{e \in E(C_s)} c_e \neq 0$ for every directed cycle C_s in G and condition (2.1) holds for every $x \in X$. Moreover, we consider that $v(x)$ is a finite value for every $x \in X$. Taking into account that the potential transformation does not change the cost of the cycles, we obtain that this transformation does not change optimal strategies of the players although values $v(x)$ of positions $x \in X$ are changed by $v(x) - \varepsilon(x) + \varepsilon(x_f)$. It is easy to observe that if we put $\varepsilon(x) = v(x)$ for $x \in X$, then the function $\varepsilon : X \to \mathbb{R}$ determines a potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ on edges $(x, y) \in E$ such that $\mathrm{ext}(c', x) = 0$, $\forall x \in X$.

$\Longleftarrow$ Let us consider that there exists a potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ on edges $(x, y) \in E$ such that $\mathrm{ext}(c', x) = 0$, $\forall x \in X$. The value $v(x)$ of the game after the potential transformation is zero for every $x \in X$ and optimal strategies of the players can be found by fixing s^*_A and s^*_B such that $s^*_A(x) \in \mathrm{VEXT}(c', x)$ for $x \in X_A \setminus \{x_f\}$ and $s^*_B(x) \in \mathrm{VEXT}(c', x)$ for $x \in X_B \setminus \{x_f\}$. Since the potential transformation does not change optimal strategies of the players we put $v(x) = \varepsilon(x) - \varepsilon(x_f)$ and obtain (2.1). $\square$

Corollary 2.5. *If for every directed cycle* C_s *in* G *the condition* $\sum_{e \in E(G_s)} c_e \neq 0$ *and equality (2.1) hold then there exists a potential transformation* $\varepsilon : X \to \mathbb{R}$ *such that* $\mathrm{ext}(c', x) = 0$*,* $\varepsilon(x_f) = 0$ *and* $v(x) = \varepsilon(x)$*,* $\forall x \in X$*.*

Corollary 2.6. *If for every directed cycle* C_s *in* G *the condition* $\sum_{e \in E(G_s)} c_e \neq 0$ *holds then the existence of a potential transformation* $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ *on edges* $(x, y) \in E$ *such that*

$$\mathrm{ext}(c', x) = 0, \forall x \in X \tag{2.4}$$

represents a necessary and sufficient condition for the validity of equality (2.1) for every $x \in X$*. In the case that in* G *there exists a cycle* C_s *with* $\sum_{e \in E(C_s)} c_e = 0$ *condition (2.4) becomes only a necessary one for the validity of equality (2.1) for every* $x \in X$*.*

Corollary 2.7. *If in a c-game there exist the strategies* s^*_A *and* s^*_B*, for which (2.1) holds for every* $x \in X$ *and these strategies generate in* G *a tree* $T_{s^*} = (X, E_{s^*})$ *with sink vertex* x_f*, then there exists a potential transformation* $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ *on edges* $(x, y) \in E$ *such that the graph* $G^0 = (X, E^0)$*, generated by the set of edges* $E^0 = \{(x, y) \in E \mid c'_{(x,y)} = 0\}$*, contains the tree* T_{s^*} *as a subgraph.*

Taking into account the results mentioned above we propose an algorithm for determining optimal strategies of the players in a c-game based on constructing of the tree of max-min paths. This algorithm works if such a tree in G exists.

2.3.4 A Polynomial Time Algorithm for Determining Optimal Strategies of the Players in a Dynamic c-Game

We consider the dynamic c-game determined by network (G, X_A, X_B, c, x_0) where the graph G has a sink vertex x_f. At first we assume that for an arbitrary vertex there exists the value $v(x)$ which satisfies condition (2.4) and $v(x) \neq \pm\infty$. So, we assume that in G there exists a tree of max-min paths from $x \in X$ to x_f. We show that for determining optimal strategies of the players in the considered game there exists a polynomial time algorithm. In this section we propose such an algorithm based on the reduction to an auxiliary dynamic c-game with an acyclic network $(\overline{G}, W_A, W_B, \overline{c}, w_f^0)$, where graph $\overline{G} = (W, \overline{E})$ is obtained from $G = (X, E)$ in the following way:

The set of vertices W consists of $n-1$ copies of vertex set X and sink vertex w_f^0, i.e.

$$W = \{w_f^0\} \cup W^1 \cup W^2 \cup \cdots \cup W^{n-1},$$

where $W^i = \{w_0^i, w_1^i, \ldots, w_{n-1}^i\}$, $i = \overline{1, n-1}$. Here $W^i \cap W^j = \emptyset$ for $i \neq j$ and vertices $w_k^i \in W^i$, $i = \overline{1, n-1}$, correspond to vertex x_k from $X = \{x_0, x_1, x_2, \ldots, x_{n-1}\}$.

The set of edges $\overline{E}$ is defined in the following way:

$$\begin{aligned}
\overline{E} &= E^0 \cup E^1 \cup E^2 \cup \cdots \cup E^{n-1};\\
E^i &= \{(w_k^{i+1}, w_l^i) \mid (x_k, x_l) \in E\}, \quad i = \overline{1, n-2};\\
E^0 &= \{(w_k^i, w_f^0) \mid (x_k, x_f) \in E,\ i = \overline{1, n-1}\}.
\end{aligned}$$

In $\overline{G}$ the edge subset $E^i \subseteq \overline{E}$ connects vertices of the set W^{i+1} with vertices of set W^i by edges (w_k^{i+1}, w_l^i) if in G there exists a directed edge (x_k, x_l). Additionally, in $\overline{G}$ each vertex w_k^i, $i = \overline{1, n-1}$, is connected with sink vertex w_f^0 by edge (w_k^i, w_f^0) if in G there exists a directed edge (x_k, x_f).

The subsets W_A, W_B and the cost function $\overline{c}$: $\overline{E} \to \mathbb{R}$ are defined as follows:

$$W_A = \{w_k^i \in W \mid x_k \in X_A\}, \quad W_B = \{w_k^i \in W \mid x_k \in X_B\};$$

$$\overline{c}(w_k^{i+1}, w_l^i) = c(x_k, x_l), \quad (x_k, x_l) \in E \ \text{ and } \ (w_k^{i+1}, w_l^i) \in E^i;\ i = \overline{1, n-2};$$

$$\overline{c}(w_k^i, w_f^0) = c(x_k, x_f), \quad (x_k, x_f) \in E \ \text{ and } \ (w_k^i, w_f^0) \in E^0;\ i = \overline{1, n-1}.$$

¿From $\overline{G}$ we delete all vertices w_k^i for which there are no directed paths from w_k^i to w_f^0. For the obtained directed graph we will preserve the same notation and we will keep in mind that $\overline{G}$ does not contain such vertices.

Let us consider the dynamic c-game determined by the acyclic network $(\overline{G},\ W_A,\ W_B,\ \overline{c},\ w_f^0)$ with sink vertex w_f^0. So, we consider the problem of determining the values $v'(w_k^i)$ of the game for every $w_k^i \in W$.

We show that if $v'(w_k^1),\ v'(w_k^2),\ \dots,\ v'(w_k^{n-1})$ are the corresponding values of vertices $w_k^1,\ w_k^2,\ \dots,\ w_k^{n-1}$ in an auxiliary game, then there exists an $i \in \{1, n-1\}$ such that $v(x_k) = v'(w_k^i)$. The vertex w_k^i we seek among $w_k^{n-1}, w_k^{n-2}, \dots, w_k^2, w_k^1$ starting with the highest level set W^{n-1}.

We consider in $\overline{G}$ the max-min path

$$P_{\overline{G}}(w_k^{n-1}, w_f^0) = \{w_k^{n-1}, w_{k_1}^{n-2}, w_{k_2}^{n-3}, \dots, w_{k_r}^{n-r-1}, w_f^0\}$$

from w_k^{n-1} to w_f^0 generated by directed edges $e = (w_{k_i}^{n-i-1}, w_{k_{i+1}}^{n-i})$ for which

$$\varepsilon'(w_{k_{i+1}}^{n-i}) - \varepsilon'(w_{k_i}^{n-i-1}) + \overline{c}(w_{k_i}^{n-i-1}, w_{k_{i+1}}^{n-i}) = 0,$$

where $\varepsilon'(w_k^j) = v'(w_k^j)$ for every $w_k^j \in \{w_k^{n-1}, w_{k_1}^{n-2}, w_{k_2}^{n-3}, \dots, w_{k_r}^{n-r-1}, w_f^0\}$. The directed path $P_{\overline{G}}(w_k^{n-1}, w_f^0)$ corresponds in G to a directed path

$$P_G(x_k, x_f) = \{x_k, x_{k_1}, x_{k_2}, \dots, x_{k_r}, x_f\}$$

from x_k to x_f. In G we consider the subgraph $G_k^{n-1} = (X_k^{n-1}, E_k^{n-1})$ induced by the set of vertices $X_k^{n-1} = \{x_k, x_{k_1}, x_{k_2}, \dots, x_{k_r}, x_f\}$. For vertices x_{k_i} and x_k we put $v(x_{k_i}) = v'(w_{k_i}^{n-i-1})$, $v(x_k) = v'(w_k^{n-1})$ and verify if in G_k^{n-1} the following condition holds:

$$\operatorname{ext}(c', z) = 0, \quad \forall z \in X_k^{n-1}, \tag{2.5}$$

where $c'(z,x) = \varepsilon(x) - \varepsilon(z) + c(z,x)$ for $e = (z,x) \in E_k^{n-1}$.

If condition (2.5) holds and G_k^{n-1} does not contain directed cycles then we may conclude that for the dynamic c-game on G with starting position x_k holds $v(x_k) = v'(w_k')$. Note that for every vertex x_{k_i} of the directed path $P_0(x_{k_i}, x_f)$ we obtain $v(x_{k_i}) = v'(w_{k_i}^{n-i-1})$. If the condition mentioned above does not take place, then $v(x_k) \neq v'(w_k^{n-1})$ and we delete w_k^{n-1} from $\overline{G}$. After that, consider vertex w_k^{n-2}, construct the graph $G_k^{n-2} = (X_k^{n-2}, E_k^{n-2})$ and in the same way verify if $v(x_k) = v'(w_k^{n-1})$. Finally, we obtain that at least for one vertex w_k^i the directed path $P_{\overline{G}}(w_k^i, w_f)$ does not contain a directed cycle and condition (2.5) holds, i.e. $v(x_k) = v(w_k^i)$. In such a way we obtain $v(x_k)$ for every $x_k \in X$.

If $v(x)$ is known for every $x \in X$ then we fix $\varepsilon(x) = v(x)$ and define a potential transformation $c'(z,x) = c(z,x) + \varepsilon(x) - \varepsilon(z)$ on the edges $(z,x) \in E$. After that, find the graph $G^0 = (V, E^0)$, generated by the set of edges $E^0 = \{(z,x) \in E \mid c'(z,x) = 0\}$. In G^0 fix an arbitrary tree $T^* = (V, E^*)$, which determines the optimal strategies of the players as follows:

$$s_A^*(z) = x, \quad \text{if} \quad (z,x) \in E^* \quad \text{and} \quad z \in X_A \setminus \{x_f\};$$

$$s_B^*(z) = x, \quad \text{if} \quad (z,x) \in E^* \quad \text{and} \quad z \in V_B \setminus \{x_f\}. \quad \square$$

The correctness of the algorithm is based on the following theorem:

Theorem 2.8. *Let $v(x_k)$ be the value of vertex x_k in the dynamic c-game on G and*

$$P_G(x_k, x_f) = \{x_k, x_{k_1}, x_{k_2}, \dots, x_{k_r}, x_f\}$$

be the max-min path from x_k to x_f in G. Then $v'(w_k^{r+1}) = v(x_k)$.

Proof. The construction described above allows us to conclude that between the set of directed paths from x_k to x_f with no more than $r+1$ edges in G and the set of directed paths from w_k^{r+1} to w_f^0 with no more than $r+1$ edges in G there exists a bijective mapping which preserves the sum of costs of the edges. Therefore, $v'(w_k^{r+1}) = v(x_k)$. □

Remark 2.9. If $P_G(x_k, x_f) = \{x_k, x_{k_1}, x_{k_2}, \dots, x_{k_r}, x_f\}$ is the max-min path from x_k to x_f in G then in $\overline{G}$ may exist several vertices $w_k^{r+i} \in W$ for which $v'(w_k^{r+i}) = v(x_k)$, where $i \geq 1$. If $v'(w_k^{r+i}) = v(x_k)$, then in $\overline{G}$ the max-min path $P_{\overline{G}}(w_k^{r+1}, w_f^0) = \{w_k^{r+i}, w_{k_1}^{r+i-1}, w_{k_2}^{r+i-2}, \dots, w_{k_r}^{i}, w_f^0\}$ corresponds to the max-min path $P_G(x_k, x_f)$ in G.

It is easy to observe that the running time of the algorithm is $O(n^4)$. Indeed, the values of the positions of the game on an acyclic network can be calculated in time $O(N^2)$, where N is the number of vertices of the network. Taking into account that $N \approx n^2$ for our auxiliary network we obtain that the running time of the algorithm is $O(n^4)$.

Note that the proposed algorithm can be also applied for the c-game when the tree of max-min paths in G may not exist but there exists a max-min path from a given vertex $x = x_0$ to x_f, i.e. the algorithm can be applied for a c-game with starting position x_0.

An important problem for a dynamic c-game is how to determine vertices $x \in X$ for which $v(x) = +\infty$ and vertices $x \in X$ for which $v(x) = -\infty$. Taking into account that the final position x_f in such games cannot be reached we may delete vertices x of the graph G for which there exist max-min paths from x to x_f. In order to specify the algorithm for this case we need to study the infinite dynamic c-game where the graph G has no sink vertex x_f. This means that the outcome of the game is a cycle which may have positive, negative or zero-sum cost of edges. For determining the outcome of the game in this case we can use the same approach based on the reduction to an acyclic c-game.

The algorithm for finding optimal strategies of the players in infinite dynamic c-games is similar to the algorithm for finding optimal strategies of the players in cyclic games. Such an algorithm we describe in Section 2.5.6 and we can see that for an arbitrary position $x \in X$ the value of the cyclic game is positive if and only if the value $v(x)$ of the infinite dynamic c-game is positive. Additionally, we can see that an efficient polynomial time algorithm for solving cyclic games can be elaborated if a polynomial time algorithm for solving an infinite dynamic c-game exists.

In the following we give an example which illustrates the details of the algorithm proposed above.

Example. Consider a dynamic c-game determined by network (G, X_A, X_B, c, x_f) given in Fig. 2.2. The position set X_A of the first player is represented by cycles and the position set X_B of the second player is represented by squares; $x_f = 0$. The costs of edges are given alongside them.

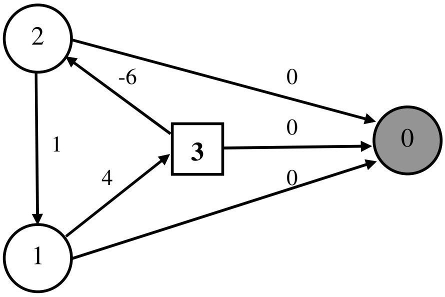

Fig. 2.2.

The auxiliary acyclic network for our dynamic c-game is represented in Fig. 2.3.

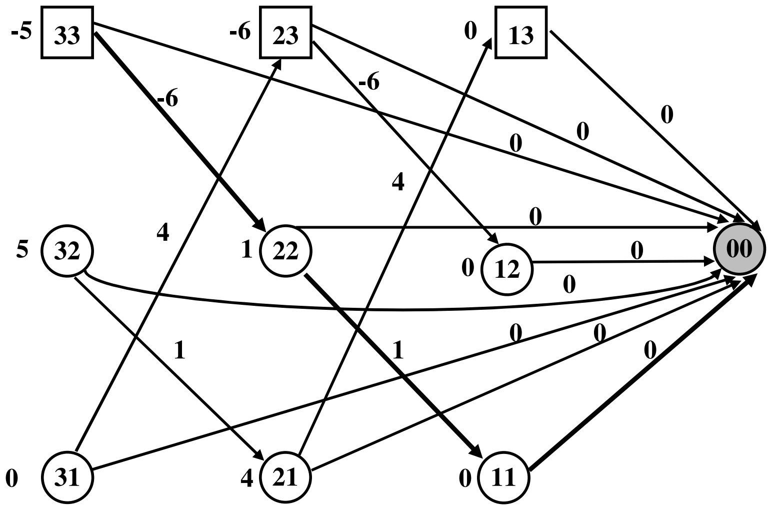

Fig. 2.3.

Each vertex in Fig. 2.3 is represented by double numbers where the first one represents the number of the copy in G and the second one corresponds to the number of the vertex in G. Alongside edges there are given their costs and alongside vertices there are given values of the dynamic c-game on an auxiliary network.

Let us fix vertex $w_k^{n-1} = 33$ as starting position of the dynamic c-game on the auxiliary network. Then we obtain $v'(33) = -5$. In order to verify if $v(3) = -5$ we find the max-min path $P_{\overline{G}}(33, 00) = \{33, 22, 11, 00\}$ and the values $v'(33) = -5$, $v'(22) = 1$, $v'(11) = 0$, $v'(00) = 0$. The path $P_{\overline{G}}(33, 00)$ in G corresponds to path $P_G(3, 2, 1, 0)$. For vertices $3, 2, 1, 0$ in G we fix $\varepsilon(3) = v'(33) = -5$, $\varepsilon(2) = v'(22) = 1$, $\varepsilon(1) = v'(11) = 0$, $\varepsilon(0) = v'(00) = 0$. After that, find graph $G_3^3 = (X_3^3, E_3^3)$ generated by the set of vertices $X^3 = \{3, 2, 1, 0\}$. Then make a potential transformation $c'(x, y) = \varepsilon(y) - \varepsilon(x) + c(x, y) = 0$ with given $\varepsilon(3) = -5,\ \varepsilon(2) = 1,\ \varepsilon(1) = 0,\ \varepsilon(0) = 0$,

$$\begin{aligned}
c'(1,0) &= \varepsilon(0) - \varepsilon(1) + c(1,0) = 0 - 0 + 0 = 0,\\
c'(2,0) &= \varepsilon(0) - \varepsilon(2) + c(2,0) = 0 - 1 + 0 = -1,\\
c'(3,0) &= \varepsilon(0) - \varepsilon(3) + c(3,0) = 0 - (-5) + 0 = 5,\\
c'(1,3) &= \varepsilon(3) - \varepsilon(1) + c(1,3) = -5 - 0 + 4 = -1,\\
c'(2,1) &= \varepsilon(1) - \varepsilon(2) + c(2,1) = 0 - 1 + 1 = 0,\\
c'(3,2) &= \varepsilon(2) - \varepsilon(3) + c(3,2) = 1 - (-5) - 6 = 0.
\end{aligned}$$

So, after the potential transformation $c'(x, y) = \varepsilon(y) - \varepsilon(x) + c(x, y)$, $\forall (x, y) \in E$, we obtain the network given in Fig. 2.4 with new costs on the edges.

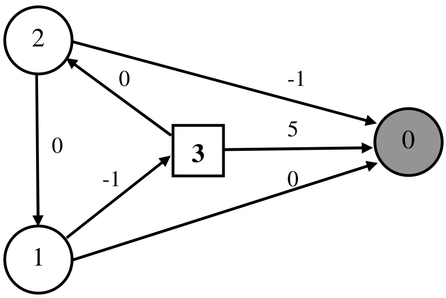

Fig. 2.4.

If we select the tree with zero-cost edges we obtain the tree of max-min paths, represented in Fig. 2.5.

If we start with vertex $w_k^{n-1} = 32$ then we obtain the subgraph $G_2^3 = (X_2^3, E_2^3)$ which coincides with the graph $G = (X, E)$ and for which $\varepsilon(2) = v'(32) = 5$, $\varepsilon(1) = v'(21) = 4$, $\varepsilon(3) = v'(13) = 0$, $\varepsilon(0) = v'(00) = 0$. It is easy to see that in this case the condition

$$\operatorname{extr}(c', x) = 0, \quad \forall x \in X,$$

is not satisfied.

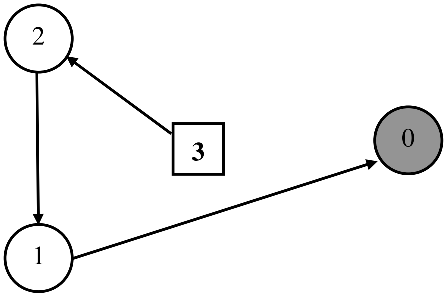

Fig. 2.5.

2.3.5 A Pseudo-Polynomial Time Algorithm for Solving a Dynamic c-Game

In this section we describe an algorithm for solving a dynamic c-game which is based on a special recursive procedure for calculating the values $v(x)$ of the game. From the practical point of view the proposed algorithm may be more useful than the algorithm from the previous section although its computational complexity is $O(|X|^3 \sum_{e \in E} |c_e|)$ ($c : E \to \mathbb{R}$ is an integer function).

We assume that in G there exists the tree of max-min paths.

Preliminary step (step 0) Set $X^* = \{x_f\}$, $\varepsilon(x_f) = 0$.

General step (step k) Find the set of vertices

$$X' = \{z \in X \setminus X^* \mid (z, x) \in E, x \in X^*\}.$$

For each $z \in X'$ we calculate

$$\varepsilon(z) = \begin{cases} \max\limits_{x \in O_{X^*}(z)} \{\varepsilon(x) + c(z,x)\}, & z \in X_A \bigcap X'; \\ \min\limits_{x \in O_{X^*}(z)} \{\varepsilon(x) + c(z,x)\}, & z \in X_B \bigcap X', \end{cases} \tag{2.6}$$

where $O_{X^*}(z) = \{x \in X^* \mid (z, x) \in E\}$, and then do the following points a) and b):

a) Fix $\beta(z) = \varepsilon(z)$ for $z \in X' \cup X^*$ and then for every $x \in X' \cup X^*$ calculate

$$\beta(z) = \begin{cases} \max\limits_{x \in O_{X^* \cup X'}(z)} \{\varepsilon(x) + c(z,x)\}, \, z \in X_A \cap (X' \cup X^*); \\ \min\limits_{x \in O_{X^* \cup X'}(z)} \{\varepsilon(x) + c(z,x)\}, \, z \in X_B \cap (X' \cup X^*). \end{cases} \tag{2.7}$$

b) Check if $\beta(z) = \varepsilon(z)$ for every $z \in X' \cup X^*$. If this condition is not satisfied then fix $\varepsilon(z) = \beta(z)$ for $z \in X' \cup X^*$; go to point a).

If $\beta(z) = \varepsilon(z)$ for every $z \in X' \cup X^*$ then in $X' \cup X^*$ we find the subset

$$Y^k = \left\{ z \in X^* \bigcup X' \;\middle|\; \mathrm{extr}_{x \in O_{X^* \cup X'}(z)} \{\varepsilon(x) - \varepsilon(z) + c(z,x)\} = 0, \right\}$$

where

$$\mathrm{ext}_{x \in O_{X^* \cup X'}(z)} \{\varepsilon(x) - \varepsilon(z) + c(z,x)\} =$$

$$= \begin{cases} \max\limits_{x \in O_{X^* \cup X'}(z)} \{\varepsilon(x) - \varepsilon(z) + c(z,x)\}, & z \in (X' \cup X^*) \cup X_A; \\ \min\limits_{x \in O_{X^* \cup X'}(z)} \{\varepsilon(x) - \varepsilon(z) + c(z,x)\}, & z \in (X' \cup X^*) \cup X_B, \end{cases}$$

After that we change X^* by Y^k and check if $X^* = X$. If $X^* \neq X$, then go to the next step. If $X^* = X$, then define a potential transformation $c'(z,x) = c(z,x) + \varepsilon(x) - \varepsilon(z)$ on the edges $(z,x) \in E$ and find the graph $G^0 = (X, E^0)$, generated by the set of edges $E^0 = \{(z,x) \in E \mid c'(z,x) = 0\}$. In G^0 fix an arbitrary tree $T^* = (X, E^*)$, which determines optimal strategies of the players as follows:

$$\begin{aligned} s_A^*(z) &= x, \qquad \text{if } \ (z,x) \in E^* \ \text{ and } \ z \in X_A \setminus \{x_f\}; \\ s_B^*(z) &= x, \qquad \text{if } \ (z,x) \in E^* \ \text{ and } \ z \in X_B \setminus \{x_f\}. \quad \square \end{aligned}$$

Let us show that this algorithm finds the tree of max-min paths $T^* = (X, E^*)$ if such a tree exists in G.

Theorem 2.10. *If in G there exists the tree of max-min paths $T^* = (X, E^*)$ with sink vertex x_f then the algorithm finds it using $O(|X|^3 \sum_{e \in E} |c_e|)$ elementary operations.*

Proof. Consider the set Y^{k-1} obtained after $k-1$ steps of the algorithm and assume that at step k after the points a) and b) the condition

$$\beta(z) = \varepsilon(z) \qquad \text{for every } \ z \in X'$$

holds. This condition is equivalent to the condition

$$\mathrm{ext}_{x \in O_{X^* \cup X'}(z)} \{\varepsilon(x) - \varepsilon(z) + c(z,x)\} = 0, \quad \forall z \in X'$$

which involves $Y^{k-1} \subset Y^k$. Therefore, in the following we obtain that if for every step k of the algorithm the corresponding calculation procedure (2.7) is convergent then $Y^0 \subset Y^1 \subset Y^2 \subset \cdots \subset Y^r = X$, where $r < n$. This means that after $r < n$ steps the algorithm finds values $\varepsilon(x)$ for $x \in X$ and a potential transformation $c'(y,x) = \varepsilon(x) - \varepsilon(y) + c(y,x)$ for edges $e = (y,x) \in E$ such that $\mathrm{ext}(c', y) = 0$, $\forall x \in X$, i.e. the algorithm constructs the tree $T^* = (X, E^*)$. So, for an complete proof of the theorem we have to show the convergence of the calculation procedure based on formula (2.7) for an arbitrary step k of the algorithm.

Assume that at step k of the algorithm the following condition

$$\mathrm{ext}_{x\in O_{X^*\cup X'}(z)}\{\varepsilon(x)-\varepsilon(z)+c(z,x)\}\neq 0 \quad \text{for every} \ \ z\in X'.$$

holds. Consider the set of edges $E'=\{e=(z,x')\in E \mid \beta(z)=\varepsilon(x')+c(z,x'),\ z\in X',\ x'\in x\in O_{X^*\cup X'}(z)\}$ where x' corresponds to vertex z such that

$$\varepsilon(x')+c(z,x')=\begin{cases}\max_{x\in O_{X^*\cup X'}(z)}\{\varepsilon(x)+c(z,x)\}, & z\in X_A\cap(X'\cup X^*);\\ \min_{x\in O_{X^*\cup X'}(z)}\{\varepsilon(x)+c(z,x)\}, & z\in X_B\cap(X'\cup X^*),\end{cases}$$

The calculation on the basis of a) and b) can be treated as follows: The players improve the values $\varepsilon(z)$ of vertices $z\in X'$ using passages from z to corresponding vertices $x'\in O_{X^*\cup X'}(z)$. At each iteration of this calculation procedure the players can improve their income by $\beta(z)-\varepsilon(z)$ units for every position $z\in X$.

Denote by $\widetilde{X}$ the subset of vertices $z'\in X'$ for which in $G'=(X',E')$ there exist directed paths from $z'\in\widetilde{X}$ to vertices from X^{k-1}. Then the improvements mentioned above of the players are possible for an arbitrary vertex $z\in\widetilde{X}$. This means that if procedure a), b) at step k is applied then after using one iteration of this procedure we obtain $\beta(z)=\varepsilon(z)$, $\forall z\in\widetilde{X}$. In the following we can see that in order to achieve $\varepsilon(z)=\beta(z)$ for the rest of the vertices $z\in X'\setminus\widetilde{X}$ it is necessary to use more than one iteration.

Let us consider in G' the subset $\widetilde{X'}=X'\setminus\widetilde{X}$. Then in G' there are no directed edges $e=(z,x')$ such that $z\in\widetilde{X'}$ and $x'\in X^{k-1}$. Without loss of generality we may consider that $\widetilde{X'}$ in G generates a directed cycle. Denote by $n(C)$ the number of vertices of this cycle and assume that the sum of its edges is equal to θ (θ may be positive or negative). We can see that if we apply formula (2.7) then after each $n(G)$ iterations of the calculation procedure the values $\varepsilon(z)$ of the vertices $z\in C$ will decrease at least by $|\theta|$ units if $\theta<0$; if $\theta>0$ then these values will increase by θ. Therefore, the first player will preserve passages from vertices $z\in C$ to vertices x' of the cycle C if $\beta(z)-\varepsilon(z)>0$; otherwise the first player will change the passage from one vertex $z^0\in C$ to a vertex $x''\in O_{X^*\cup X'}(z)$ which may belong to X^{k-1}. In an analogous way the second player will preserve passages from vertices $z\in C$ to vertices x' of cycle C if $\beta(z)-\varepsilon(z)<0$; otherwise the second player will change the passage from one vertex $z^0\in X$ to a vertex x'' which may belong to X^{k-1}. So, if in G there exists the tree of max-min paths then after a finite number of iterations of procedure a), b) we obtain $\beta(z)=\varepsilon(z)$ for $z\in X'$. Taking into account that the values $\beta(z)$ will decrease (or increase) after each $n(G)$ iteration by integer units $|\theta|$ we may conclude that the number of iterations of the procedure is comparable with $|X|^2\cdot\max_{z\in X'}|\beta(z)-\varepsilon(z)|$. In the worst case these quantities are limited by $|X|^2\sum_{e\in E}|c_e|$. This involves that the computational complexity of the algorithm is $O(|X|^3\sum_{e\in E}|c(e)|)$.□

Remark 2.11. The algorithm for acyclic networks can be applied without points a) and b), because the condition $\beta(z) = \varepsilon(z)$, $\forall z \in X'$ holds at every step k. In general, the version of the algorithm can be used without the points a) and b) if $Y^{k-1} \neq Y^k$ at every step k. In this case the running time of the algorithm is $O(|X|^3)$.

The algorithm described above can be modified for the dynamic c-game in general form when the network contains vertices x for which $v(x) = \pm\infty$. In order to detect such vertices in point a) it is necessary to introduce a new condition which allows us to select vertices $z \in X'$ with big values $\beta(z)$ (positive and negative). But in this case the algorithm becomes more difficult than the algorithm for finite games. Below we present two examples which illustrate the details of the algorithm. The first example illustrates the work of the algorithm when it is not necessary to use the points a) and b). The second example illustrates the details of the recursive calculation procedure in the points a) and b).

Example 1. Consider the problem of determining optimal stationary strategies on a network which may contain cycles. The corresponding network with sink vertex $x_f = 5$ is given in Fig. 2.6. In this network the positions of the first player are represented by circles and the positions of the second player are represented by squares, i.e. $X_1 = \{1, 2, 4, 5\}$, $X_2 = \{0, 3\}$. The values of cost functions of the edges are given in parenthesis alongside them.

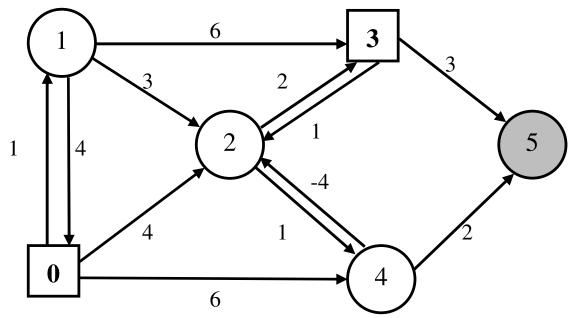

Fig. 2.6.

We can see that for this network there exists a tree of max-min paths which can be found by using the algorithm.

Step 0.
$X^* = \{5\}$; $\varepsilon(5) = 0$.

Step 1.
Find the set of vertices $X' = \{3, 4\}$ for which there exist directed edges $(3, 5)$ and $(4, 5)$ from vertices 3 and 4 to vertex 5. Then we calculate

according to (2.6) values $\varepsilon(3) = 3,\ \ \varepsilon(4) = 2$. It is easy to check that for vertices 3 and 4 the following condition holds:

$$\mathrm{ext}_{y \in X_{X^* \cup X'}(x)}\{\varepsilon(y) - \varepsilon(x) + c_{(x,y)}\} = 0.$$

So, $Y^1 = \{3, 4, 5\}$. Therefore, if we change X^* by Y^1, after step 1 we obtain $X^* = \{3, 4, 5\}$.

Step 2.
Find the set of vertices $X' = \{0, 1, 2\}$ for which there exist directed edges from vertices $x \in X'$ to vertices $y \in X^*$. Then according to (2.6) we calculate

$$\varepsilon(2) = \max_{y \in X^*(2)}\{\varepsilon(3) + c_{(2,3)}, \varepsilon(4) + c_{(2,4)}\} = \max\{5, 3\} = 5;$$

$$\varepsilon(1) = \varepsilon(3) + c_{(1,3)} = 9;$$

$$\varepsilon(0) = \varepsilon(4) + 6 = 8.$$

So, $\varepsilon(0) = 8,\ \ \varepsilon(1) = 9,\ \ \varepsilon(2) = 5,\ \ \varepsilon(3) = 3,\ \ \varepsilon(4) = 2,\ \ \varepsilon(5) = 0.$
It is easy to check that $Y^2 = \{0, 2, 3, 4, 5\}$. Indeed,

$$\begin{aligned}
&\mathrm{ext}_{y \in X_{X^* \cup X'}(3)}\{\varepsilon(y) - \varepsilon(3) + c_{(3,y)}\} = \\
&\quad = \min\{\varepsilon(5) - \varepsilon(3) + c_{(3,5)}, \varepsilon(2) - \varepsilon(3) + c_{(3,2)}\} \\
&\quad = \min\{0 - 3 + 3, 5 - 3 + 1\} = 0;
\end{aligned}$$

$$\begin{aligned}
&\mathrm{ext}_{y \in X_{X^* \cup X'}(2)}\{\varepsilon(y) - \varepsilon(3) + c_{(2,y)}\} = \\
&\quad = \max\{\varepsilon(3) - \varepsilon(2) + c_{(2,3)}, \varepsilon(4) - \varepsilon(2) + c_{(2,4)}\} \\
&\quad = \max\{3 - 5 + 2, 2 - 5 + 1\} = 0;
\end{aligned}$$

$$\begin{aligned}
&\mathrm{ext}_{y \in X_{X^* \cup X'}(1)}\{\varepsilon(y) - \varepsilon(1) + c_{(1,y)}\} = \\
&\quad = \max\{\varepsilon(3) - \varepsilon(1) + c_{(1,3)}, \varepsilon(2) - \varepsilon(1) + c_{(1,2)}, \varepsilon(0) - \varepsilon(1) + c_{(0,1)}\} \\
&\quad = \max\{3 - 9 + 6, 3 - 9 + 5, 8 - 9 + 4\} = 3;
\end{aligned}$$

$$\begin{aligned}
&\mathrm{ext}_{y \in X_{X^* \cup X'}(0)}\{\varepsilon(y) - \varepsilon(0) + c_{(0,y)}\} = \\
&\quad = \min\{\varepsilon(4) - \varepsilon(0) + c_{(0,4)}, \varepsilon(2) - \varepsilon(0) + c_{(0,2)}, \varepsilon(1) - \varepsilon(0) + c_{(0,1)}\} \\
&\quad = \min\{2 - 8 + 6, 5 - 8 + 4, 9 - 8 + 1\} = 0;
\end{aligned}$$

$$\begin{aligned}
&\mathrm{ext}_{y \in X_{X^* \cup X'}(4)}\{\varepsilon(y) - \varepsilon(4) + c_{(4,y)}\} = \\
&\quad = \max\{\varepsilon(5) - \varepsilon(4) + c_{(4,5)}, \varepsilon(2) - \varepsilon(4) + c_{(4,2)}\} \\
&\quad = \max\{0 - 2 + 2, 5 - 2 - 4\} = 0.
\end{aligned}$$

So, the set of vertices for which $\mathrm{ext}_{y \in X_{X^* \cup X'}(x)}\{\varepsilon(y) - \varepsilon(x) + c_{(x,y)}\} = 0$ consists of vertices $0, 2, 3, 4, 5$.

Step 3.
Find the set of vertices $X' = \{1\}$ and calculate

$$\begin{aligned}\varepsilon(1) &= \max_{y \in X_{X^*}(1)} \{\varepsilon(y) + c_{(1,y)}\} \\ &= \max\{\varepsilon(3) + c_{(1,3)}, \varepsilon(2) + c_{(1,2)}, \varepsilon(0) + c_{(1,0)}\} \\ &= \max\{3 + 6, 5 + 3, 9 + 4\} = 12.\end{aligned}$$

Now we can see that the obtained values $\varepsilon(0) = 8$, $\varepsilon(1) = 12$, $\varepsilon(2) = 5$, $\varepsilon(3) = 3$, $\varepsilon(4) = 2$, $\varepsilon(5) = 0$ satisfy the conditions

$$\varepsilon(y) - \varepsilon(x) + c_{(x,y)} \le 0 \quad \text{for every} \quad (x, y) \in E, x \in X_A;$$

$$\varepsilon(y) - \varepsilon(x) + c_{(x,y)} \ge 0 \quad \text{for every} \quad (x, y) \in E, x \in X_B.$$

The directed tree $GT = (X, E^*)$ generated by edges $(x, y) \in E$ for which $\varepsilon(y) - \varepsilon(x) + c_{(x,y)} = 0$ is represented in Fig. 2.7.

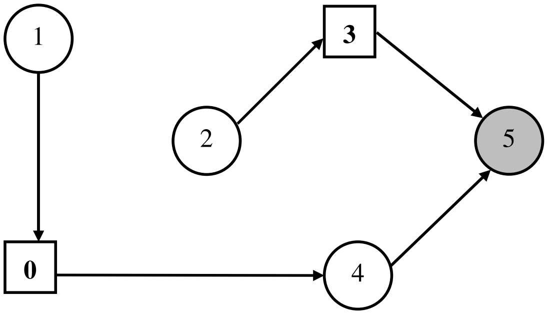

Fig. 2.7.

The optimal strategies of the players are:

$$s_A : 1 \to 0; \quad 2 \to 3; \quad 4 \to 5;$$

$$s_B : 0 \to 4; \quad 3 \to 5.$$

Example 2. Consider the problem of determining the tree of max-min paths $T^* = (X, E^*)$ for the network given in Fig. 2.3 with the same costs of edges as in the previous section.

If we apply the algorithm described above then we use only one step ($k = 1$). But at step 1 we use the points a) and b) and make calculations on the basis of formula (2.7). In Table 2 there are given values $\beta(0)$, $\beta(1)$, $\beta(2)$, $\beta(3)$ at each iteration of the procedure.

Table 2

i	$\beta(0)$	$\beta(1)$	$\beta(2)$	$\beta(3)$
0	0	0	0	0
1	0	4	1	-6
2	0	0	5	-5
3	0	0	1	-1
4	0	3	1	-5
5	0	0	4	-5
6	0	0	1	-2
7	0	2	1	-5
8	0	0	3	-5
9	0	0	1	-3
10	0	1	1	-5
11	0	0	2	-5
12	0	0	1	-4
13	0	0	1	-5
14	0	0	1	-5

We can see that the convergence of the calculation procedure is obtained at iteration 14. Therefore, we conclude that $\varepsilon(0) = 0$, $\varepsilon(1) = 0$, $\varepsilon(2) = 1$, $\varepsilon(3) = -5$. If we make a potential transformation we obtain the network in Fig. 2.4. In Fig. 2.5 it is presented the tree of max-min paths $T^* = (X, E^*)$.

2.4 A Polynomial Time Algorithm for Solving Acyclic *l*-Games on Networks

An acyclic l-game on networks has been introduced in [56, 57] as an auxiliary problem for studying and solving cyclic games, which we will consider in the next section.

2.4.1 Problem Formulation

Let (G, X_A, X_B, c) be a network, where $G = (X, E)$ represents a directed acyclic graph with sink vertex $x_f \in X$. On E it is defined a function $c\colon E \to \mathbb{R}$ and on X it is given a partition $X = X_A \cup X_B$ $(X_A \cap X_B = \emptyset)$ where X_A and X_B correspond to positions sets of the two players A and B, respectively.

We consider the following acyclic game from [56]. Again we define strategies of the players as maps

$$s_A\colon x \to y \in X(x) \quad \text{for} \quad x \in X_A \setminus \{x_f\};$$
$$s_B\colon x \to y \in X(x) \quad \text{for} \quad x \in X_B \setminus \{x_f\}.$$

We define a payoff function $\overline{H}_{x_0}\colon S_A \times S_B \to \mathbb{R}$ in this game as follows:

Let $s_A \in S_A$ and $s_B \in S_B$ be fixed strategies of the players. Then the graph $G_s = (X, E_s)$, generated by edges $(x, s_A(x))$, $x \in X \setminus \{x_f\}$, and $(x, s_B(x))$, $x \in X \setminus \{x_f\}$, has the structure of a directed tree with sink vertex x_f. Therefore, it contains a unique directed path $P_s(x_0, x_f)$ with $n(P_s(x_0, x_f))$ edges. We put

$$\overline{H}_{x_0}(s_A, s_B) = \frac{1}{n(P_s(x_0, x_f))} \sum_{e \in E(P_s(x_0, x_f))} c_e.$$

The payoff function $\overline{H}_{x_0}(s_A, s_B)$ on $S_A \times S_B$ defines a game in normal form, which is determined by network $(G, X_A, X_B, c, x_0, x_f)$.

We consider the problem of finding strategies s_A^* and s_B^*, for which

$$\overline{v}(x_0) = \overline{H}_{x_0}(s_A^*, s_B^*) = \max_{s_A \in S_A} \min_{s_B \in S_B} \overline{H}_{x_0}(s_A, s_B).$$

2.4.2 Main Properties of Optimal Strategies in Acyclic *l*-Games

First of all let us show that for the considered max-min problem there exists a saddle point.

Denote

$$\overline{\overline{v}}(x_0) = \overline{H}_{x_0}(s_A^0, s_B^0) = \min_{s_B \in S_B} \max_{s_A \in S_A} \overline{H}_{x_0}(s_A, s_B)$$

and let us show that $\overline{v}(x_0) = \overline{\overline{v}}(x_0)$.

Theorem 2.12. *For an arbitrary acyclic l-game the following equality holds:*

$$\overline{v}(x_0) = \overline{H}_{x_0}(s_A^*, s_B^*) = \max_{s_A \in S_A} \min_{s_B \in S_B} \overline{H}_{x_0}(s_A, s_B) = \min_{s_B \in S_B} \max_{s_A \in S_A} \overline{H}_{x_0}(s_A, s_B).$$

Proof. First of all let us note the following property of an acyclic l-game, determined by $(G, X_A, X_B, c, x_0, x_f)$: If the cost function c is changed by $c' = c + h$ (h is an arbitrary real number), then we obtain an equivalent acyclic l-game determined by $(G, X_A, X_B, c', x_0, x_f)$ for which $\overline{v}'(x_0) = \overline{v}(x_0) + h$ and $\overline{\overline{v}}''(x_0) = \overline{\overline{v}}(x_0) + h$. It is easy to observe that if $h = -\overline{v}(x_0)$ then for the acyclic l-game with network $(G, X_A, X_B, c', x_0, x_f)$ we obtain $\overline{v}'(x_0) = 0$. This means that the acyclic l-game becomes an acyclic c-game for which the following property holds:

$$0 = \overline{v}'(x_0) = \max_{s_A \in S_A} \min_{s_B \in S_B} \overline{H}'_{x_0}(s_A, s_B) = \min_{s_B \in S_B} \max_{s_A \in S_A} \overline{H}'_{x_0}(s_A, s_B) = 0.$$

Taking into account that

$$\overline{H}'_{x_0}(s_A, s_B) = \overline{H}_{x_0}(s_A, s_B) - \overline{v}(x_0)$$

we obtain that

$$\min_{s_B \in S_B} \max_{s_A \in S_A} \left(\overline{H}_{x_0}(s_A, s_B) - \overline{v}(x_0) \right) =$$

$$= \max_{s_A \in S_A} \min_{s_B \in S_B} \left(\overline{H}_{x_0}(s_A, s_B) - \overline{v}(x_0) \right) = \overline{\overline{v}}(x_0) - \overline{v}(x_0),$$

i.e. $\overline{\overline{v}}(x_0) - \overline{v}(x_0) = 0$. So, $\overline{\overline{v}}(x_0) = \overline{v}(x_0)$. □

Theorem 2.13. *Let an acyclic l-game determined by network $(G, X_A, X_B, c, x_0, x_f)$ with starting position x_0 be given. Then there exist a value $\overline{v}(x_0)$ and a function $\varepsilon \colon X \to \mathbb{R}$, which determine a potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(x) - \varepsilon(y)$ of costs on the edges $e = (x, y) \in E$ such that the following conditions hold*

a) $\overline{v}(x_0) = \mathrm{ext}(c', x), \ \forall x \in X \setminus \{x_f\}$;

b) $\varepsilon(x_0) = \varepsilon(x_f)$.

The optimal strategies of the players in an acyclic l-game can be found as follows: Fix the arbitrary maps $s^*_A \colon X_A \setminus \{x_f\} \to X$ and $s^*_B \colon X_B \setminus \{x_f\} \to X$ such that $s^*_A(x) \in \mathrm{VEXT}(c', x)$ for $x \in X_A \setminus \{x_f\}$ and $s^*_B(x) \in \mathrm{VEXT}(c', x)$ for $x \in X_B \setminus \{x_f\}$.

Proof. The proof of the theorem follows from Theorem 2.1 if we regard the acyclic l-game as an acyclic c-game on network $(G, X_A, X_B, c', x_0, x_f)$ with cost function $c' = c - \overline{v}(x_0)$. □

Corollary 2.14. *The difference $\varepsilon(x) - \varepsilon(x_0)$, $x \in X$, represents the costs of a max-min path from x to x_f in an acyclic c-game on network $(G, X_A, X_B, c', x_0, x_f)$ with $c'_{(x,y)} = c_{(x,y)} - \overline{v}(x_0), \ \forall \ (x, y) \in E$.*

2.4.3 A Polynomial Time Algorithm for Finding the Value and the Optimal Strategies in an Acyclic *l*-Game

The algorithm, which we describe below, is based on the results from Section 2.4.2. In this algorithm we shall use the following properties:

1. The value $\overline{v}(x_0)$ of an acyclic l-game on network $(G, X_A, X_B, c, x_0, x_f)$ is nonnegative if and only if the value $v(x_0)$ of an acyclic l-game on network $(G, X_A, X_B, c, x_0, x_f)$ is nonnegative; moreover $\overline{v}(x_0) = 0$ if and only if $v(x_0) = 0$.
2. If $M^1 = \min_{e \in E} c_e$ and $M^2 = \max_{e \in E} c_e$, then $M^1 \le \overline{v}(x_0) \le M^2$.
3. If in network $(G, X_A, X_B, c, x_0, x_f)$ the cost function $c : E \to \mathbb{R}$ is changed by the function $c^h : E \to \mathbb{R}$, where

$$c^h_e = c_e - h, \quad \forall e \in E \tag{2.8}$$

(h is an arbitrary constant), then the acyclic l-games on $(G, X_A, X_B, c, x_0, x_f)$ and $(G, X_A, X_B, c^h, x_0, x_f)$, respectively, have the same optimal strategies s^*_A, s^*_B. Additionally, the values $\overline{v}(x_0)$ and $\overline{v}_h(x_0)$ of these games differ by a constant h: $\overline{v}_h(x_0) = \overline{v}(x_0) - h$. So, the acyclic l-games on $(G, X_A, X_B, c, x_0, x_f)$ and $(G, X_A, X_B, c^h, x_0, x_f)$ are equivalent.

According to the properties mentioned above, if $\overline{v}(x_0)$ is known, then the acyclic l-game can be reduced to an acyclic c-game by using transformation (2.3) with $h = \overline{v}(x_0)$. After that, we can find optimal strategies in the game with network $(G, X_A, X_B, c^h, x_0, x_f)$ by using Algorithm 2.2. The most important aspect for us in the proposed algorithm is represented by the problem of finding a value h, for which $\overline{v}_h(x_0) = 0$. Taking into account properties 1 and 2, we will seek for this value by using a dichotomy method on segment $[M^1, M^2]$, such that at each step of this method we will solve a dynamic c-game with network $(G, X_A, X_B, c^k, x_0, x_f)$, where $c^k = c - h_k$. The main idea of the general step of the algorithm is the following: We make transformation (2.8) with $h = h_k$, where h_k is the midpoint of segment $[M_k^1, M_k^2]$ at step k. After that, we apply Algorithm 2.2 for the dynamic c-game on network $(G, X_A, X_B, c^{h_k}, x_0, x_f)$ and find $v_{h_k}(x_0)$. If $v_{h_k}(x_0) > 0$ then we fix segment $[M_{k+1}^1, M_{k+1}^2]$, where $M_{k+1}^1 = M_k^1$ and $M_{k+1}^2 = \frac{M_k^1 + M_k^2}{2}$; otherwise we put $M_{k+1}^1 = \frac{M_k^1 + M_k^2}{2}$ and $M_{k+1}^2 = M_k^2$. If $v_{h_k}(x_0) = 0$ then STOP. The detailed description of the algorithm is the following:

Algorithm 2.15. **Determining the Value and the Optimal Strategies in an Acyclic l-Game**

Let us assume that the cost function $c : E \to \mathbb{R}$ is an integer and $\max_{e \in E} |c_e| \neq 0$.

Preliminary step (step 0): Find value $v(x_0)$ and optimal strategies s_A^* and s_B^* of the dynamic c-game on $(G, X_A, X_B, c, x_0, x_f)$ by using Algorithm 2.2. If $v(x_0) = 0$ then fix s_A^* and s_B^* as solution of the l-game, put $\overline{v}(x_0) = 0$ and STOP; otherwise fix $M_1^1 = \min_{e \in E} c_e$, $M_1^2 = \max_{e \in E} c_e$, $L = \max_{e \in E} |c_e| + 1$.

General step (step k, $k \geq 1$): Find $h_k = \frac{M_k^1 + M_k^2}{2}$ and make a transformation of the edges' costs

$$c_e^k = c_e - h_k \quad \text{for} \quad e \in E.$$

Solve the dynamic c-game on network $(G, X_A, X_B, c^k, x_0, x_f)$ and find value $v_k(x_0)$ and optimal strategies s_A^*, s_B^*. If $v_k(x_0) = 0$ then fix the optimal strategies s_A^* and s_B^* and put $\overline{v}(x_0) = h_k$. If $|v_k(x_0)| \leq \frac{1}{4|X|^2 L}$ then fix s_A^* and s_B^*; find $\overline{v}(x_0) = \frac{\overline{H}_{x_0}(s_A^*, s_B^*)}{n(P_{s^*}(x_0, x_f))}$ and STOP. If $v_k(x_0) > \frac{1}{4|X|^2 L}$ then fix $M_{k+1}^1 = M_k^1$, $M_{k+1}^2 = h_k$ and go to step $k+1$. If $v_k(x_0) < -\frac{1}{4|X|^2 L}$ then fix $M_{k+1}^1 = h_k$, $M_{k+1}^2 = M_k^2$ and go to step $k+1$.

Theorem 2.16. *Let* $(G, X_A, X_B, c, x_0, x_f)$ *be a network with integer cost function* $c : E \to \mathbb{R}$, *and* $L = \max_{e \in E} |c_e|$. *Then Algorithm 2.15 finds correctly value* $\overline{v}(x_0)$ *and optimal strategies* s_A^*, s_B^* *in the acyclic* l*-game. The running time of the algorithm is* $O(|X|^2 \log L + 2|X|^2 \log |X|)$.

Proof. Let $(G, X_A, X_B, c^k, x_0, x_f)$ be a network after final step k of Algorithm 2.15. Then

$$|v_k(x_0)| \le \frac{1}{4|X|^2 L}$$

and the number $\varepsilon_k(x)$, $x \in X$, determined according to Algorithm 2.2 (when we solve the acyclic c-game), represents an approximative solution of the system

$$\begin{cases} \varepsilon(y) - \varepsilon(x) + c^k_{(x,y)} \le 0 \text{ for } x \in X_A, \ (x,y) \in E; \\ \varepsilon(y) - \varepsilon(x) + c^k_{(x,y)} \ge 0 \text{ for } x \in X_B, \ (x,y) \in E; \\ \varepsilon(x_0) = \varepsilon(x_f). \end{cases}$$

This means that $\varepsilon_k(x)$, $x \in X$, and h_k represent an approximative solution of the system

$$\begin{cases} \varepsilon(y) - \varepsilon(x) + c_{(x,y)} \le h \text{ for } x \in X_A, \ (x,y) \in E; \\ \varepsilon(y) - \varepsilon(x) + c_{(x,y)} \ge h \text{ for } x \in X_B, \ (x,y) \in E; \\ \varepsilon(x_0) = \varepsilon(x_f). \end{cases}$$

According to [45, 46], the exact solution $h = \overline{v}(x)$, $\varepsilon(x)$, $x \in X$, of this system can be obtained from h_k, $\varepsilon_k(x)$, $x \in X$, by using the special round-off procedure in time $O(\log(L+1))$. Therefore, the strategies s_A^*, s_B^* after final step k of the algorithm correspond to the optimal solution of the acyclic l-game.

Taking into account that the tabulation of values $\varepsilon(x)$, $x \in X$, in G needs $O(|X|^2)$ operations and the number of iterations of the algorithm is $O(\log L + 2 \log |X|)$, we obtain that the running time of the algorithm is $O(|X|^2 \log L + 2|X|^2 \log |X|)$. □

2.5 Cyclic Games: Algorithms for Finding the Value and the Optimal Strategies of the Players

Cyclic games have been introduced in [22, 40, 79] as extension of control models for discrete systems with infinite time horizon and mean integral-time cost by a trajectory. Here we show that the problem of finding optimal strategies of the players in such games is tightly connected with the problem of finding optimal strategies of players in a dynamic c-game and an acyclic l-game. On the basis of these results we propose algorithms for determining value and optimal strategies in cyclic games.

2.5.1 Problem Formulation and Main Properties

Let $G = (X, E)$ be a finite directed graph in which every vertex $x \in X$ has at least one leaving edge $e = (x, y) \in E$. On edge set E it is given a function c: $E \to \mathbb{R}$ which assigns a cost c_e to each edge $e \in E$. Additionally, the vertex set X is divided into two disjoint subsets X_A and X_B ($X = X_A \cup X_B,\ X_A \cap X_B = \emptyset$) which we will regard as positions sets of the two players.

On G we consider the following two-person game from [22, 40, 105, 107]: The game starts at position $x_0 \in X$. If $x_0 \in X_A$ then the move is done by the first player, otherwise it is done by the second one. Move means the passage from position x_0 to neighbor position x_1 through edge $e_1 = (x_0, x_1) \in E$. After that, if $x_1 \in X_A$ then the move is done by the first player, otherwise it is done by the second player and so on indefinitely. The first player has the aim to maximize $\lim_{t\to\infty} \inf \frac{1}{t}\sum_{i=1}^{t} c_{e_i}$ while the second player has the aim to minimize $\lim_{t\to\infty} \sup \frac{1}{t}\sum_{i=1}^{t} c_{e_i}$.

In [22] it is proved that for this game there exists a value $\overline{v}(x_0)$ such that the first player has a strategy of moves that insures $\lim_{t\to\infty} \inf \frac{1}{t}\sum_{i=1}^{t} c_{e_i} \geq \overline{v}(x_0)$ and the second player has a strategy of moves that insures $\lim_{t\to\infty} \sup \frac{1}{t}\sum_{i=1}^{t} c_{e_i} \leq \overline{v}(x_0)$. Furthermore, in [22] it is shown that the players can achieve value $\overline{v}(x_0)$ applying the strategies of moves which do not depend on t. This means that the considered game can be formulated in terms of stationary strategies. Such a characterization statement of the game in [40] is named cyclic game.

The strategies of the players in a cyclic game are defined as maps

$$s_A\colon x \to y \in X(x) \quad \text{for} \quad x \in X_A,$$
$$s_B\colon x \to y \in X(x) \quad \text{for} \quad x \in X_B,$$

where $X(x) = \{y \in X \mid e = (x, y) \in E\}$. Since G is a finite graph then the sets of strategies of the players

$$S_A = \{s_A\colon x \to y \in X(x) \ \text{ for } \ x \in X_A\};$$
$$S_B = \{s_B\colon x \to y \in X(x) \ \text{ for } \ x \in X_B\}$$

are finite sets. The payoff function $\overline{H}_{x_0}\colon S_A \times S_B \to \mathbb{R}$ in the cyclic game is defined as follows:

Let $s_A \in S_A$ and $s_B \in S_B$ be fixed strategies of the players. Denote by $G_s = (X, E_s)$ the subgraph of G generated by edges of the form $(x, s_A(x))$ for $x \in X_A$ and $(x, s_B(x))$ for $x \in X_B$. Then G_s contains a unique directed cycle C_s which can be reached from x_0 through the edges $e \in E_s$. We assume that the value $\overline{H}_{x_0}(s_A, s_B)$ is equal to the mean edges cost of cycle C_s, i.e.

$$\overline{H}_{x_0}(s_A, s_B) = \frac{1}{n(C_s)} \sum_{e \in E(C_s)} c_e,$$

where $E(C_s)$ represents the set of edges of cycle C_s and $n(C_s)$ is the number of edges of C_s. So, the cyclic game is determined uniquely by network

(G, X_A, X_B, c, x_0), where x_0 is a given starting position of the game. If we consider the problem of finding optimal strategies of the players for an arbitrary starting position $x \in X$, then we will use the notation (G, X_A, X_B, c). In [22, 40] it is proved that there exist strategies $s_A^* \in S_A$ and $s_B^* \in S_B$ such that

$$\overline{v}(x) = \overline{H}_x(s_A^*, s_B^*) = \max_{s_A \in S_A} \min_{s_B \in S_B} \overline{H}_x(s_A, s_B)$$

$$= \min_{s_B \in S_B} \max_{s_A \in S_A} \overline{H}_x(s_A, s_B), \quad \forall x \in X.$$

So, the optimal strategies s_A^*, s_B^* of the players in cyclic games do not depend on a starting position x_0 although for different positions $x, y \in X$ the values $\overline{v}(x)$ and $\overline{v}(y)$ may be different. This means that the positions set X can be divided into several classes $X = X^1 \cup X^2 \cup \cdots \cup X^k$ according to values of positions $\overline{v}^1, \overline{v}^2, \ldots, \overline{v}^k$, i.e. $x, y \in X^i$ if and only if $\overline{v}^i = \overline{v}(x) = \overline{v}(y)$. In case $k = 1$ the network (G, X_A, X_B, c) is named ergodic network [40]. In [55, 60] it is shown that every cyclic game with an arbitrary network (G, X_A, X_B, c, x_0) and a given starting position x_0 can be reduced to a cyclic game on an auxiliary ergodic network (G', X'_A, X'_B, c').

It is well-known [44, 105, 107] that the decision problem associated to a cyclic game is in $NP \bigcap$ co-NP. Some exponential and pseudo-polynomial algorithms for finding value and optimal strategies of the players in a cyclic game are proposed in [107]. Our aim is to propose polynomial time algorithms for determining optimal strategies of the players in cyclic games. We discuss such algorithms on the basis of the results which have been announced in [60, 61].

2.5.2 Determining the Best Response of the First Player for a Fixed Strategy of the Second Player

In order to find the best response of the first player for a fixed strategy of the second player we shall use the model from Section 2.5.1 in case $X_B = \emptyset$, i.e. $X = X_A$. This case of the model corresponds to the problem of finding in G the maximal mean cost cycle, which can be reached from x_0. An efficient polynomial time algorithm for finding a maximal mean cost cycle in an weighted directed graph is proposed in [14, 43]. In [55, 98, 99] it is shown that for a strongly connected graph this problem can be represented as the following linear programming problem:
Maximize the objective function

$$\overline{H} = \sum_{e \in E} c_e \alpha_e$$

subject to

$$\begin{cases} \sum_{e\in E^-(x)} \alpha_e - \sum_{e\in E^+(x)} \alpha_e = 0, \quad \forall x \in X; \\ \sum_{e\in E} \alpha_e = 1; \\ \alpha_e \ge 0, \quad e \in E, \end{cases}$$

where $E^-(x)$ is a set of edges $e = (y,x) \in E$, which have their extremities in x, and $E^+(x)$ is a set of edges $e = (x,y) \in E$, originated in x. The variable α_e is associated with each edge $e \in E$.

An arbitrary admissible solution α of the considered linear programming problem determines in G a flow circulation with a constant (equal to 1) sum of flow values by the edges of the directed weighted graph G. It is easy to show, that any admissible solution of the linear programming problem can be represented in the form of a convex combination of flow values of elementary directed cycles with a constant (equal to 1) sum of flow values by edges of these cycles. Thus, associating to each solution α of a polyhedral admissible set Z_α of the problem the directed subgraph $G_\alpha = (X_\alpha, E_\alpha)$ generated by the edges $e \in E$ with $\alpha_e > 0$, we obtain that any of the extreme points α' of the polyhedral set Z_α will correspond to the subgraph $G_{\alpha'}$ of G, which has the structure of an elementary directed cycle. So, the following lemma holds:

Lemma 2.17. *If α' is a solution of the problem, which corresponds to an extreme point of Z_α, then the graph $G_{\alpha'}$ represents an elementary cycle in G.*

On the basis of this lemma in [55, 98] the following theorem is proved:

Theorem 2.18. *If α^* is an optimal basic solution of the considered linear programming problem, then the cycle C_{α^*} is the maximal mean cost cycle in G.*

So, the problem of finding the maximal mean cost cycle in G can be solved by using the polynomial algorithm. Moreover, on the basis of duality theory, we can find a condition for determining value v of the maximal mean cycle, and the solution. Indeed, if for our linear programming problem we define the dual problem:

Minimize

$$z = v$$

subject to

$$\varepsilon(x) - \varepsilon(y) + v \ge c_{(x,y)}, \quad \forall (x,y) \in E,$$

then we obtain the following result which is similar to the one from [8]:

Theorem 2.19. *For a given strongly connected directed graph $G = (X, E)$ there exist a value v and a function $\varepsilon : X \to \mathbb{R}$ such that*

$$c'_{(x,y)} = \varepsilon(y) - \varepsilon(x) + c_{(x,y)} - v \le 0, \quad \forall (x, y) \in E$$

and

$$\max_{y \in X(x)} c'_{(x,y)} = 0, \quad \forall x \in X.$$

Moreover, if we fix in G an arbitrary map $s^* : x \to y \in X(x)$ such that $c'_{(x,s(x))} = 0$, $\forall x \in X$, then an arbitrary directed cycle C in $G_{\alpha^*} = (X, E_{\alpha^*})$ is a solution of the problem.

Let us show that if $\overline{s}_B$ is an arbitrary fixed strategy of the second player then the best response $\overline{s}^*_A$ of the first player can be found by using the approach described above. Indeed, if the second player fixes his strategy $\overline{s}_B$ then this means that in G the set of edges $E_{\overline{s}_B} = \{(x, \overline{s}_B(x)) \mid x \in X_B\}$ is fixed. Therefore, we obtain a subgraph $\overline{G} = (X, \overline{E})$, where $\overline{E} = E_A \cup E_{s_B}$ and $E_A = \{(x, y) \in E \mid x \in X_A\}$, and in order to obtain the best response of the first player we have to find in this graph the maximal mean cost cycle, which corresponds to a solution

$$\overline{s}^*_A : \overline{H}_x(\overline{s}^*_A, \overline{s}_B) = \max_{s_A} \overline{H}_x(s_A, \overline{s}_B) \quad \text{for } \forall x \in X.$$

This approach based on the alternate best response of the players in cyclic games, of course, can be used for solving some classes of cyclic games. But such an approach can not be estimated from the computational point of view. Therefore, in the following we will propose another approach for determining optimal strategies in cyclic games.

A similar continuous model can be used for determining the best response of the first player for a fixed strategy of the second player in the acyclic l-game. If we consider $X = X_A$ then the optimal mean directed path from starting position x_0 to final position x_f can be found on basis of the following linear programming problem:

Maximize the objective function

$$\overline{H}_{x_0} = \sum_{e \in E} c_e \alpha_e$$

subject to

$$\begin{cases} \displaystyle\sum_{e \in E^-(x)} \alpha_e - \sum_{e \in E^+(x)} \alpha_e = \begin{cases} -1, & x \ne x_0; \\ \;\;\,0, & x \ne x_0, x_f; \\ \;\;\,1, & x \ne x_f; \end{cases} \\ \displaystyle\sum_{e \in E} \alpha_e = 1; \\ \alpha_e \ge 0, \quad e \in E. \end{cases}$$

2.5.3 Some Preliminary Results

First of all we need to remind some preliminary results from [40, 55, 56, 60, 61].

Let (G, X_A, X_B, c) be a network with the properties described in Section 2.5.1. In an analogous way as for dynamic c-games here we denote

$$\operatorname{ext}(c,x) = \begin{cases} \max\limits_{y \in X(x)} c_{(x,y)} & \text{for } \ x \in X_A, \\ \min\limits_{y \in X(x)} c_{(x,y)} & \text{for } \ x \in X_B, \end{cases}$$

$$\operatorname{VEXT}(c,x) = \{y \in X(x) \mid c_{(x,y)} = \operatorname{ext}(c,x)\}.$$

We shall use the potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ for costs on the edges $e = (x,y) \in E$, where $\varepsilon\colon X \to \mathbb{R}$ is an arbitrary function on the vertex set X. In [40] it is noted that the potential transformation does not change value and optimal strategies of the players in cyclic games.

Theorem 2.20. *Let (G, X_A, X_B, c) be an arbitrary network with the properties described in Section 2.5.1. Then there exists a value $\overline{v}(x)$, $x \in X$ and a function $\varepsilon\colon X \to \mathbb{R}$ which determines a potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ for costs on edges $e = (x,y) \in E$, such that the following properties hold:*

a) $\overline{v}(x) = \operatorname{ext}(c', x)$ *for* $x \in X$,

b) $\overline{v}(x) = \overline{v}(y)$ *for* $x \in X_A \cup X_B$ *and* $y \in \operatorname{VEXT}(c', x)$,

c) $\overline{v}(x) \geq \overline{v}(y)$ *for* $x \in X_A$ *and* $y \in X_G(x)$,

d) $\overline{v}(x) \leq \overline{v}(y)$ *for* $x \in X_B$ *and* $y \in X_G(x)$,

e) $\max_{e \in E} |c'_e| \leq 2|X| \max_{e \in E} |c_e|$.

The values $\overline{v}(x), x \in X$ on network (G, X_A, X_B, c) are determined uniquely and optimal strategies of the players can be found in the following way: Fix arbitrary strategies $s_A^*\colon X_A \to X$ and $s_B^*\colon X_B \to X$ such that $s_A^*(x) \in \operatorname{VEXT}(c', x)$ for $x \in X_A$ and $s_B^*(x) \in \operatorname{VEXT}(c', x)$ for $x \in X_B$.

The proof of Theorem 2.20 is given in [40]. Another proof of Theorem 2.20 can be obtained if the conditions of Theorem 2.19 are applied with respect to each position set of the players.

Furthermore, we shall use Theorem 2.20 in the case of a ergodic network (G, X_1, X_2, c), i.e. we shall use the following corollary:

Corollary 2.21. *Let (G, X_A, X_B, c) be an ergodic network. Then there exists a value $\overline{v}$ and a function $\varepsilon\colon X \to \mathbb{R}$ which determines a potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ for costs of edges $e = (x,y) \in E$ such that $\overline{v} = \operatorname{ext}(c', x)$ for $x \in X$. The optimal strategies of the players can be found as follows: Fix arbitrary strategies $s_A^*\colon X_A \to X$ and $s_B^*\colon X_B \to X$ such that $s_A^*(x) \in \operatorname{VEXT}(c', x)$ for $x \in X_A$ and $s_B^*(x) \in \operatorname{VEXT}(c', x)$ for $x \in X_B$.*

2.5.4 The Reduction of Cyclic Games to Ergodic Games

Let us consider an arbitrary network (G, X_A, X_B, c) with a given starting position $x_0 \in X$ which determines a cyclic game. In [55, 60, 61] it is shown that this game can be reduced to a cyclic game on an auxiliary ergodic network $(G', W_A, W_B, \overline{c})$, $G' = (W, E')$ with the same value $\overline{v}(x_0)$ of the game as the initial one, where $x_0 \in W = X \cup U \cup Z$.

The graph $G' = (W, E')$ is obtained from G if each edge $e = (x, y)$ is changed by a triple of edges $e^1 = (x, u), e^2 = (u, z), e^3 = (z, y)$ with the costs $\overline{c}_{e^1} = \overline{c}_{e^2} = \overline{c}_{e^3} = c_e$. Here $u \in U$, $z \in Z$ and $x, y \in X$; $W = X \cup U \cup Z$. Additionally, in G' each vertex u is connected with x_0 by edge (u, x_0) with the cost $\overline{c}_{(u,x_0)} = M$ (M is a big value) and each vertex z is connected with x_0 by edge (z, x_0) with the cost $\overline{c}_{(z,x_0)} = -M$. In $(G', W_A, W_B, \overline{c})$ the sets W_A and W_B are defined as follows: $W_A = X_A \cup Z$; $W_B = X_B \cup U$.

It is easy to observe that this reduction can be done in linear time.

2.5.5 A Polynomial Time Algorithm for Solving Ergodic Zero-Value Cyclic Games

Let us consider an ergodic zero-value cyclic game determined by a network (G, X_A, X_B, c, x_0), where $G = (X, E)$. Then according to Theorem 2.20 there exists a function $\varepsilon : X \to \mathbb{R}$ which determines a potential transformation $c'_{(x,y)} = c_{(x,y)} + \varepsilon(y) - \varepsilon(x)$ on edges $(x, y) \in E$ such that

$$\mathrm{ext}(c, x) = 0, \quad \forall x \in X. \tag{2.9}$$

This means that if x_f is a vertex of the cycle C_{s^*} determined by optimal strategies s_A^* and s_B^* then the problem of finding a function $\varepsilon : X \to \mathbb{R}$ which determines a canonic potential transformation is equivalent to the problem of finding values $\varepsilon(x)$, $x \in X$ in a max-min paths problem on G with sink vertex x_f where $\varepsilon(x_f) = 0$.

So, in order to solve the zero-value cyclic game each time we fix a vertex $x \in X$ as a sink vertex ($x_f = x$) and solve a max-min paths problem on G with sink vertex x_f. If for a given $x_f = x$ a function $\varepsilon : X \to \mathbb{R}$, obtained on the basis of the algorithm from Section 2.3.4 and 2.3.5, determines a potential transformation which satisfies (2.9) then we fix s_A^* and s_B^* such that $s_A^*(x) \in \mathrm{VEXT}(c', x)$ for $x \in X_A$ and $s_B^*(x) \in \mathrm{VEXT}(c', x)$ for $x \in X_B$. If for a given x the function $\varepsilon : X \to \mathbb{R}$ does not satisfy (2.9) then we select another vertex $x \in X$ as a sink vertex and so on. This means that optimal strategies of the players in zero-value ergodic cyclic games can be determined in time $O(|X|^4)$.

Example. Consider the ergodic zero-sum cyclic game determined by a network given in Fig. 2.8 with starting position $x_0 = 0$. Positions of the first player are represented by circles and positions of the second player are represented by squares, i.e. $X_1 = \{1, 2, 4, 5\}$, $X_2 = \{0, 3\}$. The network in Fig. 2.8 is obtained from the network in Fig. 2.8 by adding edge $(5, 2)$ with cost $c_{(5,2)} = -5$. It is easy to check that the value of the cyclic game on this network for an arbitrary fixed starting position is equal to zero.

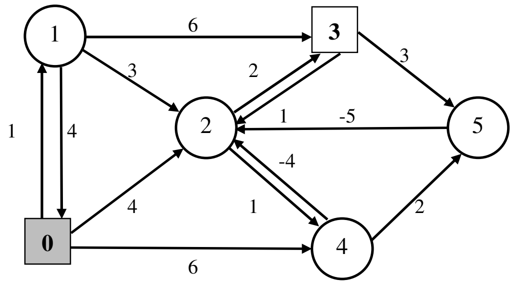

Fig. 2.8.

The max-min mean cycle which determines a way with zero-cost is $2 \to 3 \to 5 \to 2$. Therefore, if we fix a vertex of this cycle as a sink vertex (for example $x = 5$) then we can find a potential function $\varepsilon : X \to \mathbb{R}$ which determines a potential transformation $c'_{(x,y)} = \varepsilon(y) - \varepsilon(x) + c_{(x,y)}$ such that $\text{ext}(c', x) = 0,\ \forall x \in X$. This function $\varepsilon : X \to \mathbb{R}$ can be found by using the algorithm from the example in the Sections 2.3.4 and 2.3.5, i.e. we find costs of min-max paths from every $x \in X$ to vertex 5. So, $\varepsilon(0) = 8$, $\varepsilon(1) = 12$, $\varepsilon(2) = 5$, $\varepsilon(3) = 3$, $\varepsilon(4) = 2$, $\varepsilon(5) = 0$. After the potential transformation we obtain a network with the following costs of edges:

$$c'_{(3,5)} = \varepsilon(5) - \varepsilon(3) + c_{(3,5)} = 0 - 3 + 3 = 0$$

$$c'_{(4,5)} = \varepsilon(5) - \varepsilon(4) + c_{(4,5)} = 0 - 2 + 2 = 0$$

$$c'_{(5,2)} = \varepsilon(2) - \varepsilon(5) + c_{(5,2)} = 5 - 0 - 5 = 0$$

$$c'_{(2,3)} = \varepsilon(3) - \varepsilon(2) + c_{(2,3)} = 3 - 5 + 2 = 0$$

$$c'_{(3,2)} = \varepsilon(2) - \varepsilon(3) + c_{(3,2)} = 5 - 3 + 1 = 3$$

$$c'_{(4,2)} = \varepsilon(2) - \varepsilon(4) + c_{(4,2)} = 5 - 2 - 4 = -1$$

$$c'_{(2,4)} = \varepsilon(4) - \varepsilon(2) + c_{(2,4)} = 2 - 5 + 1 = -2$$

$$c'_{(0,4)} = \varepsilon(4) - \varepsilon(0) + c_{(0,4)} = 2 - 8 + 6 = 0$$

$$c'_{(0,2)} = \varepsilon(2) - \varepsilon(0) + c_{(0,2)} = 5 - 8 + 4 = 1$$

$$c'_{(1,3)} = \varepsilon(3) - \varepsilon(1) + c_{(1,3)} = 3 - 12 + 6 = -3$$

$$c'_{(1,2)} = \varepsilon(2) - \varepsilon(1) + c_{(1,2)} = 5 - 12 + 3 = -4$$

$$c'_{(1,0)} = \varepsilon(0) - \varepsilon(1) + c_{(1,0)} = 8 - 12 + 4 = 0$$

$$c'_{(0,1)} = \varepsilon(1) - \varepsilon(0) + c_{(0,1)} = 12 - 8 + 1 = 5$$

The network after potential transformation is given in Fig. 2.9. We can

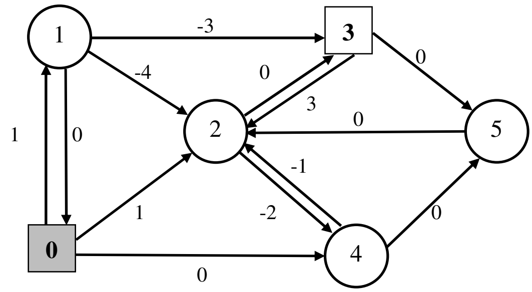

Fig. 2.9.

see that $\mathrm{ext}(c', x) = 0$, $\forall x \in X$. Therefore, the edges with zero-cost determine the optimal strategies of the players

$$s_A^* : 1 \to 0; \quad 2 \to 3; \quad 4 \to 5; \quad 5 \to 2;$$

$$s_B^* : 0 \to 4; \quad 3 \to 5.$$

The graph $G_{s^*} = (X, E_{s^*})$ generated by these strategies is represented in Fig. 2.10.

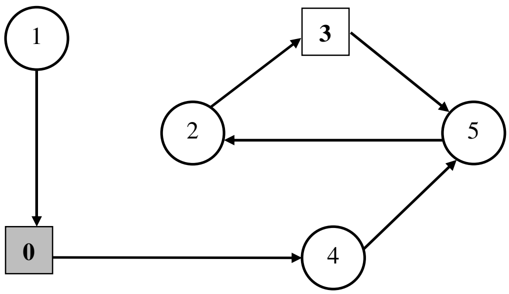

Fig. 2.10.

2.5.6 A Polynomial Time Algorithm for Solving Cyclic Games Based on the Reduction to Acyclic l-Games

On the basis of the obtained results we can propose a polynomial time algorithm for solving cyclic games.

We consider an acyclic game on a ergodic network (G, X_A, X_B, c, x_0) with a given starting position x_0. The graph $G = (X, E)$ is considered to be strongly connected and $X = \{x_0, x_1, x_2, \ldots, x_{n-1}\}$. Assume that x_0 belongs to cycle C_{s^*} determined by optimal strategies of the players s_A^* and s_B^*. If in G there are several such cycles we consider one of them with the minimum number of edges.

We construct an auxiliary acyclic graph $GT_r = (\overline{W}_r, \overline{E}_r)$, where

$$\overline{W}_r = \{w_0^0\} \cup W^1 \cup W^2 \cup \cdots \cup W^r, \quad W^i \cap W^j = \emptyset,\ i \neq j;$$

$$W^i = \{w_0^i, w_1^i, \ldots, w_{n-1}^i\}, \quad i = \overline{1, r};$$

$$\overline{E}_r = E^0 \cup E^1 \cup E^2 \cup \cdots \cup E^{r-1};$$

$$E^i = \{(w_k^{i+1}, w_l^i) \mid (x_k, x_l) \in E\}, \quad i = \overline{1, r-1};$$

$$E^0 = \{(w_k^i, w_0^0) \mid (x_k, x_0) \in E,\ i = \overline{1, r}\}.$$

The vertex set $\overline{W}_r$ of GT_r is obtained from X if it is doubled r times and a sink vertex w_0^0 is added. The edge subset $E^i \subseteq \overline{E}$ in GT_r connects vertices of the set W^{i+1} and vertices of the set W^i in the following way: If in G there exists an edge $(x_k, x_l) \in E$ then in GT_r we add the edge (w_k^{i+1}, w_l^i). The edge subset $E^0 \subseteq \overline{E}$ in GT_r connects vertices $w_k^i \in W^1 \cup W^2 \cup \cdots \cup W^r$ with sink vertex w_0^0, i.e. if there exists an edge $(x_k, x_0) \in E$ then in GT_r we add the edges $(w_k^i, w_0^0) \in E^0$, $i = \overline{1, r}$.

After that, we define an acyclic network $(GT_r', W_A, W_B, c', w_0^0)$, $GT_r' = (W_r, E_r)$ where GT_r' is obtained from GT_r by deleting the vertices $w_k^i \in \overline{W}_r$ from which vertex w_0^0 is not attainable. The sets W_A, W_B and the cost function $c'\colon E_r \to \mathbb{R}$ are defined as follows:

$$W_A = \{w_k^i \in W_r \mid x_k \in X_A\}, \quad W_B = \{w_k^i \in W_r \mid x_k \in X_B\};$$

$$c'_{(w_k^{i+1}, w_l^i)} = c_{(x_k, x_l)} \quad \text{if } \ (x_k, x_l) \in E \ \text{ and } \ (w_k^{i+1}, w_l^i) \in E^i;\ i = \overline{1, r-1};$$

$$c'_{(w_k^i, w_0^0)} = c_{(x_k, x_0)} \quad \text{if } \ (x_k, x_0) \in E \ \text{ and } \ (w_k^i, w_0^0) \in E^0;\ i = \overline{1, r}.$$

Now we consider an acyclic c-game on tan acyclic network $(GT_r', W_A, W_B, c', w_0^r, w_0^0)$ with sink vertex w_0^0 and starting position w_0^r.

Lemma 2.22. *Let $\overline{v} = \overline{v}(x_0)$ be the value of a ergodic cyclic game on G and the number of edges of the max-min cycle C_{s^*} in G is equal to r. Additionally, let $\overline{v}_r(w_0^r)$ be the value of the l-game on (GT_r', W_A, W_B, c') with starting position w_0^r. Then $\overline{v}(x_0) = \overline{v}_r(w_0^r)$.*

Proof. It is evident that there exists a bijective mapping between the set of cycles with no more than r edges (which contains the vertex x_0) in G and the set of directed paths with no more than r edges from w_0^r to w_0^0 in GT_r'. Therefore, $\overline{v}(x_0) = \overline{v}_r(w_0^r)$. □

On the basis of this lemma we can propose the following algorithm for finding optimal strategies of the players in cyclic games:

Algorithm 2.23. **Determining the Optimal Stationary Strategies of the Players in Cyclic Games with Known Vertex $\mathbf{x_0}$ of a Max-min Cycle of the Network**

We construct the acyclic networks (GT'_r, W_A, W_B, c'), $r = 2, 3, \ldots, n$, and for each of them we solve the l-game. In such a way we find the values $\overline{v}_2(w_0^2)$, $\overline{v}_3(w_0^3)$, $\ldots$, $\overline{v}_n(w_0^n)$ for these l-games. Then we fix consecutively $\overline{v} = \overline{v}_2(w_0^2)$, $\overline{v}_3(w_0^3)$, $\ldots$, $\overline{v}_n(w_0^n)$ and each time solve the c-game on network (G, X_A, X_B, c''), where $c'' = c - \overline{v}$. Fixing each time the values $\varepsilon'(x_k) = v(x_k)$ for $x_k \in X$ we check if the following condition

$$\mathrm{ext}(c^r, x_k) = 0, \quad \forall x_k \in X$$

is satisfied, where $c^r_{(x_k, x_l)} = c''_{(x_k, x_l)} + \varepsilon(x_l) - \varepsilon(x_k)$. We determine r for which this condition holds and fix the respective maps s_A^* and s_B^* such that $s_A^*(x_k) \in \mathrm{VEXT}(c'', x_k)$ for $x_k \in X_A$ and $s_B^*(x_k) \in \mathrm{VEXT}(c'', x_k)$ for $x_k \in X_B$. So, s_A^* and s_B^* represent optimal strategies of the players in the cyclic games on G.

Remark 2.24. Algorithm 2.23 finds value $\overline{v}(x_0)$ and optimal strategies of the players in time $O(|X|^5 \log L + 4|X|^3 \log |X|)$, because Algorithm 2.15 needs $O(|X|^4 \log L + 4|X|^2 \log |X|)$ elementary operations for solving the acyclic l-game on network (GT'_r, W_A, W_B, c'), where $L = \max_{e \in E} |c_e| + 1$.

In the general case, if the belonging of x_0 to the max-min cycle is unknown then we use the following algorithm:

Algorithm 2.25. **Determining the Optimal Strategies of the Players in Ergodic Cyclic Games (General Case)**

Preliminary step (step 0): Fix $Y_1 = X$.

General step (step k): Select a vertex $y \in Y_1$, fix $x_0 = y$ and apply Algorithm 2.23. If there exists $r \in \{2, 3, \ldots, n\}$ such that $\mathrm{ext}(c^r, x) = 0$, $\forall x \in X$, then fix $s_A^* \in \mathrm{VEXT}(c^k, x)$ for $x \in X_A$ and $s_B^* \in \mathrm{VEXT}(c^k, x)$ for $x \in X_B$ and STOP; otherwise put $Y_{k+1} = Y_k \setminus \{y\}$ and go to next step $k + 1$.

Remark 2.26. Algorithm 2.25 finds value $\overline{v}$ and optimal strategies of the players in time $O(|X|^6 \log L + 4|X|^4 \log |X|)$, because in the worst case Algorithm 2.23 is repeated $|X|$ times.

The algorithm for solving cyclic games allows us to determine the sign of value $v(x_0)$ in an infinite dynamic c-game on G with starting position x_0. In order to determine $\mathrm{sign}(v(x_0))$ we solve on G the cyclic game with starting position x_0 and find $\overline{v}(x_0)$. Then $\mathrm{sign}(v(x_0))$=$\mathrm{sign}(\overline{v}(x_0))$.

2.5.7 An Approach for Solving Cyclic Games Based on a Dichotomy Method and Solving Dynamic c-Games

In this section we describe an approach for solving cyclic games considering that there exist efficient algorithms for solving dynamic c-games (including infinite dynamic c-games).

Consider an ergodic cyclic game determined by an ergodic network $(G, X_A,$ $, X_B,$ $c, x_0)$ where the value of the game may be different from zero. The graph G is assumed to be strongly connected.

At first we show how to determine value of the game and optimal strategies of the players in the case that vertex x_0 belongs to a max-min cycle in G induced by optimal strategies of the players.

To our ergodic cyclic game we associate a dynamic c-game determined by an auxiliary network $(\overline{G}, X_A, X_B \cup \{x_0'\}, \overline{c}, x_0, x_0')$, where the graph $\overline{G} = (X \cup \{x_0'\}, \overline{E})$ is obtained from G by adding a copy x_0' of vertex x_0 together with copies $e' = (x, x_0')$ of edges $e = (x, x_0) \in E$ with costs $\overline{c}_{e'} = c_e$. So, for x_0' there are no leaving edges (x_0', x).

It is evident that if the value $\overline{v} = \overline{v}(x_0)$ of the ergodic cyclic game on (G, X_A, X_B, c, x_0) is known then the problem of finding optimal strategies of the players is equivalent to the problem of finding optimal strategies of the players in a dynamic c-game on network $(\overline{G}, X_A, X_B \cup \{x_0'\}, \overline{c}', x_0, x_0')$ with the cost functions

$$\overline{c}_e' = \overline{c}_e - \overline{v}(x_0) \quad \text{for} \quad e \in \overline{E}.$$

Moreover, if s_A^* and s_B^* are optimal strategies of the players in the dynamic c-game on $(\overline{G}, X_A, X_B \cup \{x_0'\}, \overline{c}', x_0, x_0')$, then optimal strategies $\overline{s}_A^*$ and $\overline{s}_B^*$ of the players in the ergodic cyclic game can be found as follows:

$$\overline{s}_A^*(x) = s_A(x) \quad \text{for} \quad x \in X_A \quad \text{if} \quad s_A(x) \neq x_0';$$

$$\overline{s}_B^*(x) = s_B(x) \quad \text{for} \quad x \in X_B \quad \text{if} \quad s_B(x) \neq x_0';$$

and

$$\overline{s}_A^*(x) = x_0 \quad \text{if} \quad s_A(x) = x_0';$$

$$\overline{s}_B^*(x) = x_0' \quad \text{if} \quad s_B(x) = x_0'.$$

It is easy to observe that for the considered problems the following properties hold:

1. The value $\overline{v}(x_0)$ of the ergodic cyclic game on network (G, X_A, X_B, c, x_0) is nonnegative if and only if the value $v(x_0)$ of the dynamic c-game on network $(\overline{G}, X_A, X_B \cup \{x_0'\}, \overline{c}, x_0, x_0')$ is nonnegative; moreover $\overline{v}(x_0) = 0$ if and only if $v(x_0) = 0$.
2. If $M^1 = \min\limits_{e \in E} c_e$ and $M^2 = \max\limits_{e \in E} c_e$, then $M^1 \leq \overline{v}(x_0) \leq M^2$.

3. If in the network (G, X_A, X_B, c, x_0) the cost function $c : E \rightarrow R$ is changed by $c' = c + h$, then the optimal strategies of players in the ergodic cyclic game on the network (G, X_A, X_B, c', x_0) do not change although the value $\overline{v}(x_0)$ is changed by $\overline{v}'(x_0) = \overline{v}(x_0) + h$.

On the basis of these properties we seek for the unknown value $\overline{v}(x_0) = v(x)$, which we denote by h, using the dichotomy method on segment $[M^1, M^2]$ such that at each step of this method we will solve a dynamic c-game with network $(\overline{G}, X_A, X_B \cup \{x_0'\}, \overline{c}^h, x_0, x_0')$, where $\overline{c}^h = \overline{c} - h$. So, the main idea of the general step of the algorithm is the following: We make a transformation

$$\overline{c}^k = \overline{c} - h_k \quad \text{for} \;\; e \in E,$$

where h_k is midpoint of segment $[M_k^1, M_k^2]$ at step k. After that we apply the algorithm from Section 2.3.5 for the dynamic c-game on network $(\overline{G}, X_A, X_B \cup \{x_0'\}, \overline{c}^k, x_0, x_0')$ and find $v_{h_k}(x_0)$. If $v_{h_k}(x_0) > 0$ then we fix segment $[M_{k+1}^1, M_{k+1}^2]$, where $M_{k+1}^1 = M_k^1$ and $M_{k+1}^2 = \frac{M_k^1 + M_k^2}{2}$; otherwise we put $M_{k+1}^1 = \frac{M_k^1 + M_k^2}{2}$ and $M_{k+1}^2 = M_k^2$. If $v_{h_k}(x_0) = 0$ then STOP.

So, using a dichotomy method in an analogous way as for an acyclic l-game we determine the value of the acyclic game. If this value of the dynamic c-game is known then we determine strategies of the players by using the algorithms from Section 2.3.4 or Section 2.3.5.

In the case that x_0 may not belong to the max-min cycle determined by the optimal strategies of the players in a cyclic game we solve $|X|$ problems by fixing each time a starting position $x_0 = x$ for $x \in X$. Then at least for one position $x_0 = x \in X$ we obtain the value of the cyclic game and the optimal strategies of the players.

2.6 Cyclic Games with Random States' Transitions of the Dynamical System

In cyclic games with random states' transitions of a dynamical system the network consisting of sets of states X_A and X_B of the players A and B also contains states $x \in X_D$ for which on the set of transitions $E^+(x)$ it is given a distribution function

$$\Pi(x, y) \geq 0 \text{ for } y \in X_G(x), \qquad \sum_{y \in X_G(x)} \Pi(x, y) = 1,$$

i.e. $\Pi(x, y)$ represents the probability of system L to pass from state x to state $y \in X_G(x)$. We denote the network by $(G, X_A, X_B, X_D, c, x_0)$, where X_A is the set of positions of the first player, X_B is the set of positions of the second player and X_D represents the set of positions with random states' transitions of the dynamical system.

This stochastic game model comprises the cyclic game from Section 2.3 ($X_D = \emptyset$), the Marcov process with income ($X_A = \emptyset, X_B = \emptyset$) [42] and the control Marcov chains with income ($X_B = \emptyset$) [42].

The payoff function $\overline{H}_{x_0} : S_A \times S_B \to \mathbb{R}$ in the cyclic game with random states' transitions of the dynamical system is defined as follows:

Let

$$s_A : x \to y \in X_G(x) \quad \text{for} \quad x \in X_A;$$

$$s_B : x \to y \in X_G(x) \quad \text{for} \quad x \in X_B$$

be arbitrary stationary strategies of the players A and B, respectively. If the players A and B fix their strategies then we can consider that they use moves on the set of transitions $E^+(x)$ with probabilities

$$\Pi(x,y) = \begin{cases} 1, \text{ if } y = s_A(x) \\ 0, \text{ if } y \neq s_A(x) \end{cases} \quad \text{for} \ \ x \in X_A;$$

$$\Pi(x,y) = \begin{cases} 1, \text{ if } y = s_B(x) \\ 0, \text{ if } y \neq s_B(x) \end{cases} \quad \text{for} \ \ x \in X_B.$$

This means that we have a full stochastic case, i.e. we have a Marcov process with income, where the probabilities $\Pi(x,y)$ and the costs for arbitrary states' transitions are given. It is well known that for such a process there exists the mean income and we denote this income by $\overline{H}_{x_0}(s_A, s_B)$.

If for the given version of a stochastic game we denote

$$\text{ext}(c,x) = \sum_{y \in X_G(x)} \Pi(x,y)\ c(x,y); \ p(x) = \sum_{y \in X_G(x)} \Pi(x,y)p(y),$$

then Theorem 1.9 holds. This involves that

$$\max_{s_A \in S_A} \min_{s_B \in S_B} \overline{H}_{x_0}(s_A, s_B) = \min_{s_B \in S_B} \max_{s_A \in S_A} \overline{H}_{x_0}(s_A, s_B).$$

2.7 A Nash Equilibria Condition for Cyclic Games with p Players

A cyclic game with p players is determined by a network $(G, X_1, X_2, \ldots, X_p, c^1, c^2, \ldots, c^p, x_0)$, where $G = (X, E)$ is a directed graph in which every vertex $x \in X$ has at least one leaving edge $e = (x, y) \in E$. A partition $X = X_1 \cup X_2 \cup \cdots \cup X_p$ $(X_i \cap X_j = \emptyset, \ i \neq j)$ on the vertex set X is given and p functions $c^1 : E \to \mathbb{R}^1$; $c^2 : E \to \mathbb{R}^1$; $\ldots$; $c^p : E \to \mathbb{R}^1$ on the edge set E are defined. The strategies of the players

$$s_i : x \to y \in X_G(x) \quad \text{for} \ \ x \in X_i, \ i = \overline{1,p}$$

and the payoff functions $\overline{H}^i_{x_0} : S_1 \times S_2 \times \cdots \times S_P \to \mathbb{R}$, $i = \overline{1,p}$, in the cyclic game with p players are defined in an analogous way as for the zero-sum cyclic game from Section 2.5. Denote by $G_s = (X, E_s)$ a subgraph of G, generated by fixed strategies $s_1, s_2, \dots, s_p$ of the players $1, 2, \dots, p$. Then G_s contains a unique directed cycle C_s, which can be reached from a given starting position x_0 through the edges $e \in E_s$. The values $\overline{H}^i_{x_0}(s_1, s_2, \dots, s_p)$ are considered to be equal to the mean edges' costs of cycle C_s, i.e.

$$\overline{H}^i_{x_0}(s_1, s_2, \dots, s_p) = \frac{1}{n(C_s)} \sum_{e \in E(C_s)} c^i_e,$$

where $n(C_s)$ is the number of edges of cycle C_s and $E(C_s)$ is the set of edges of this cycle.

In the considered game we are seeking for a Nash equilibrium, i.e. it is necessary to find strategies $s^*_1, s^*_2, \dots, s^*_p$, for which

$$\begin{aligned} \overline{H}^i_{x_0}(s^*_1, s^*_2, \dots, s^*_{i-1}, s^*_i, s^*_{i+1}, \dots, s^*_p) \le \\ \overline{H}^i_{x_0}(s^*_1, s^*_2, \dots, s^*_{i-1}, s_i, s^*_{i+1}, \dots, s^*_p) \\ \forall s_i \in S_i, \quad i = \overline{1,p}. \end{aligned} \tag{2.10}$$

Intuitively, it is clear that for cyclic games with p players Nash equilibria may not exist. An example, for which Nash equilibria in a cyclic game of two players (with maximum criteria) do not exist, is given in [40]. This example is related to a cyclic game on a complete bipartite graph $G = (X_1 \cup X_2, E)$ with set of positions $X_1 = \{x_1, x_2, x_3\}$ of the first player and set of positions $X_2 = \{y_1, y_2, y_3\}$ of the second player; $E = \{(x_i, y_j) \mid i = \overline{1,3},\ j = \overline{1,3}\}$. The cost functions of the players on the edges (in both directions) are defined by the matrices

$$C^1 = \begin{pmatrix} 0 & 0 & 1 \\ \varepsilon & 0 & 0 \\ 0 & \varepsilon & 0 \end{pmatrix}; \quad C^2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1-\varepsilon & 0 & 1 \end{pmatrix}$$

If ε is a small value (for example $\varepsilon = 0.1$) then a Nash equilibrium for such a game does not exist.

Here we formulate a necessary and sufficient condition for the existence of Nash equilibria in so-called ergodic cyclic games with p players, which extend zero-sum ergodic cyclic games.

Definition 2.27. *Let $s_1^*, s_2^*, \dots, s_p^*$ be a solution in the sense of Nash for a cyclic game determined by network $(G, X_1, X_2, \dots, X_p, c^1, c^2, \dots, c^p, x_0)$, where $G = (X, E)$ is a strongly connected directed graph. We call this game an ergodic cyclic game if $s_1^*, s_2^*, \dots, s_p^*$ represents the solution in the sense of Nash for a cyclic game on network $(G, X_1, X_2, \dots, X_p, c^1, c^2, \dots, c^p, x)$ with an arbitrary starting position $x \in X$ and*

$$\overline{H}_x^i(s_1^*, s_2^*, \dots, s_p^*) = \overline{H}_y^i(s_1^*, s_2^*, \dots, s_p^*), \quad \forall x, y \in X, \ i = \overline{1, p}.$$

Theorem 2.28. *The dynamic c-game determined by the network $(G, X_1, X_2, , \dots, X_p, c^1, c^2, \dots, c^p, x_0)$, where $G = (X, E)$ is a strongly connected directed graph, is ergodic one if and only if on X there exist p real functions*

$$\varepsilon^1 : X \to \mathbb{R}^1, \ \varepsilon^2 : X \to \mathbb{R}^1, \dots, \ \varepsilon^p : X \to \mathbb{R}^1,$$

and p values $\overline{v}^1, \overline{v}^2, \dots, \overline{v}^p$ such that the following conditions are satisfied:

a) $\varepsilon^i(x) - \varepsilon^i(y) + c^i_{(x,y)} - \overline{v}^i \ge 0, \quad \forall (x, y) \in E_i,$
where $E_i = \{e = (x, y) \in E \mid x \in X_i\}, \ i = \overline{1, p};$

b) $\min_{y \in X_G(x)}\{\varepsilon^i(x) - \varepsilon^i(y) + c^i_{(x,y)} - \overline{v}^i\} = 0, \quad \forall x \in X_i, \ i = \overline{1, p};$

c) the subgraph $\overline{G}^0 = (X, \overline{E}^0)$ generated by edge set $\overline{E}^0 = E_1^0 \cup E_2^0 \cup \cup \dots \cup E_p^0$, $E_i^0 = \{e = (x, y) \in E_i \mid \varepsilon^i(x) - \varepsilon^i(y) + c^i_{(x,y)} - \overline{v}^i = 0\}$, $i = \overline{1, p}$, has the property that it contains a connected subgraph $\overline{G}^0 = (X, \overline{E}^0)$, for which every vertex $x \in X$ has only one leaving edge $e = (x, y) \in \overline{E}^0$ and besides that

$$\varepsilon^i(x) - \varepsilon^i(y) + c^i_{(x,y)} - \overline{v}^i = 0, \quad \forall (x, y) \in \overline{E}^0, \ i = \overline{1, p}.$$

The optimal solution of the problem can be determined by fixing the maps:

$$\begin{aligned} s_1^* &: x \to y \in X_{\overline{G}^0}(x) \quad \textit{for} \quad x \in X_1; \\ s_2^* &: x \to y \in X_{\overline{G}^0}(x) \quad \textit{for} \quad x \in X_2; \\ &\vdots \\ s_p^* &: x \to y \in X_{\overline{G}^0}(x) \quad \textit{for} \quad x \in X_p, \end{aligned}$$

where $X_{\overline{G}^0}(x) = \{y \mid (x, y) \in \overline{E}^0\}$.

Proof. $\Longrightarrow$ Let $(G, X_1, X_2, \dots, X_p, c^1, c^2, \dots, c^p, x_0)$ be a network which determines an ergodic cyclic game with p players, i.e. in this game there exists a Nash equilibrium $s_1^*, s_2^*, \dots, s_p^*$ which satisfies condition (2.10). Define

$$q^i = H_{x_0}^i(s_1^*, s_2^*, \dots, s_p^*), \quad i = \overline{1,p}. \tag{2.11}$$

It is easy to verify that if we change the cost function c^i by $\overline{c}^i = c^i - q^i$, $i = \overline{1,p}$, then the obtained network $(G, X_1, X_2, \dots, X_p, \overline{c}^1, \overline{c}^2, \dots, \overline{c}^p, x_0)$ determines a new ergodic cyclic game which is equivalent to the initial one.

For the new game $s_1^*, s_2^*, \dots, s_p^*$ is a Nash equilibrium and

$$H_{x_0}^i(s_1^*, s_2^*, \dots, s_p^*) = 0, \quad i = \overline{1,p}.$$

This game can be regarded as the dynamic c-game from [5, 56] on network $(G, X_1, X_2, \dots, X_p, \overline{c}^1, \overline{c}^2, \dots, \overline{c}^p, x_0, x_0)$ with given starting position x_0 and final position $x_0 \in C_{s^*}$, where C_{s^*} is a directed cycle generated by strategies $s_1^*, s_2^*, \dots, s_p^*$ such that

$$\sum_{e \in E(C_{s^*})} \overline{c}_e^i = 0, \quad i = \overline{1,p}.$$

Taking into account that our game is ergodic we may state, without loss of generality, that x_0 belongs to a directed cycle C_{s^*} generated by strategies $s_1^*, s_2^*, \dots, s_p^*$. Therefore, the ergodic game with network $(G, X_1, X_2, \dots, X_p,$ $, \overline{c}^1, \overline{c}^2, \dots, \overline{c}^p, x_0)$ can be regarded as the dynamic c-game from [5, 56] on network $(G, X_1, X_2, \dots, X_p, \overline{c}^1, \overline{c}^2, \dots, \overline{c}^p, x_0, x_0)$ with starting position x_0 and final position x_0. So, according to Theorem 2 from [5] there exist p real functions

$$\varepsilon^1 : X \to \mathbb{R}^1, \ \varepsilon^2 : X \to \mathbb{R}^1, \dots, \ \varepsilon^p : X \to \mathbb{R}^1,$$

such that the following conditions are satisfied:

1) $\varepsilon^i(x) - \varepsilon^i(y) + \overline{c}_{(x,y)}^i \geq 0, \quad \forall (x,y) \in E_i, \ i = \overline{1,p};$
2) $\min_{x \in X_G(x)} \{\varepsilon^i(x) - \varepsilon^i(y) + \overline{c}_{(x,y)}^i\} = 0, \quad \forall x \in X_i, \ i = \overline{1,p};$
3) the subgraph $G^0 = (X, E^0)$, generated by the edge set $E^0 = E_1^0 \cup E_2^0 \cup \dots \cup E_p^0$, $E_i^0 = \{e = (x,y) \in E_i \mid \varepsilon^i(x) - \varepsilon^i(y) + \overline{c}_{(x,y)}^i = 0\}$, $i = \overline{1,p}$, has the property that it contains a connected subgraph $\overline{G}^0 = (X, \overline{E}^0)$, for which every vertex $x \in X$ has only one leaving edge $e = (x,y) \in \overline{E}^0$ and besides that
$$\varepsilon^i(x) - \varepsilon^i(y) + \overline{c}_{(x,y)}^i = 0, \quad \forall (x,y) \in \overline{E}^0, \ i = \overline{1,p}.$$

If in the conditions 1), 2) and 3) mentioned above we take into account that $\overline{c}_{(x,y)}^i = \overline{c}_{(x,y)}^i - q^i$, $\forall (x,y) \in E$, $i = \overline{1,p}$, then we obtain the conditions a), b) and c) from Theorem 2.28.

$\Longleftarrow$ Assume that the conditions a), b) and c) of Theorem 2.28 hold. Then for network $(G, X_1, X_2, \dots, X_p, \overline{c}^1, \overline{c}^2, \dots, \overline{c}^p, x_0)$ the conditions 1), 2) and 3) are satisfied. It is easy to check that an arbitrary set of strategies $s_1^*, s_2^*, \dots, s_p^*$, where

$$s_i^* : x \to y \in X_{\overline{G}^0}(x), \quad i = \overline{1,p},$$

is a Nash equilibrium for an ergodic cyclic game on network $(G, X_1, X_2, \dots, X_p, \overline{c}^1, \overline{c}^2, \dots, \overline{c}^p, x_0)$ and

$$\overline{H}^i_{x_0}(s_1^*, s_2^*, \dots, s_p^*) = 0, \quad i = \overline{1,p}.$$

This involves that $s_1^*, s_2^*, \dots, s_p^*$ determine a Nash equilibrium for the ergodic cyclic game on network $(G, X_1, X_2, \dots, X_p, c^1, c^2, \dots, c^p, x_0)$. □

Remark 2.29. The value $\overline{v}^i$, $i = \overline{1,p}$, coincides with the value of the payoff function $\overline{H}^i_x(s_1^*, s_2^*, \dots, s_p^*)$, $i = \overline{1,p}$. If $\overline{v}^i = 0$, then the ergodic cyclic game coincides with the dynamic c-game on network $(G, X_1, X_2, \dots, X_p, c^1, c^2, \dots, c^p, x_0, x_0)$.

Note that for ergodic zero-sum games Theorem 2.28 becomes a necessary and sufficient condition of the existence of Nash equilibria, i.e. we obtain Theorem 2.20.

Some extension of cyclic games for stochastic cases has been considered in [16, 36, 42].

2.8 Determining Pareto Optima for Cyclic Games with p Players

To determine a Pareto solution for a cyclic game with p players we can use the continuous model from Section 2.5.2 and extend it for the multi-objective case of the problem in the following way:
Minimize the vector function

$$\overline{H}(\alpha) = \left(\overline{H}^1(\alpha), \overline{H}^2(\alpha), \dots, \overline{H}^p(\alpha)\right)$$

subject to

$$\begin{cases} \displaystyle\sum_{e \in E^-(x)} \alpha_e - \sum_{e \in E^+(x)} \alpha_e = 0, \quad \forall x \in X; \\ \displaystyle\sum_{e \in E} \alpha_e = 1; \\ \alpha_e \geq 0, \quad e \in E, \end{cases}$$

where

$$\overline{H}^i(\alpha) = \sum_{e \in E} c_e^i \alpha_e, \quad i = \overline{1,p};$$

$$E^-(x) = \{e = (y, x) \mid (y, x) \in E\}; \quad E^+(x) = \{e = (x, y) \mid (x, y) \in E\}.$$

Pareto optima for this multi-criteria problem can be found by using the approach from [10, 11, 12, 23, 97]. Solutions of this continuous problem will correspond to solutions of the discrete multi-criteria problem on a given strongly connected graph $G = (X, E)$ with cost functions $c^i : E \to \mathbb{R}$, $i = \overline{1, p}$.

Note that a Pareto solution for the cyclic game with p players on G does not depend on the partition $X = X_1 \cup X_2 \cup \dots \cup X_p$.

http://www.springer.com/978-3-540-85024-3

Optimization and Multiobjective Control of Time-Discrete Systems
Dynamic Networks and Multilayered Structures
Lozovanu, D.
2009, XVI, 285 p. 53 illus., Hardcover
ISBN: 978-3-540-85024-3