Contributions to Game Theory and Management, XII, 37–48

Random Search Methods for the Solution of a Stackelberg Game of Resource Allocation[*]

Grigory I. Belyavsky and Natalya V. Danilova

I. I. Vorovich Institute of Mathematics,
Mechanics and Computer Sciences of Southern Federal University,
8a, Milchakova, Rostov-on-Don, Russia
beliavsky@hotmail.com
danilova198686@mail.ru

Abstract We consider a dynamic Stackelberg game on a finite time interval. The game is reduced to a problem of infinite-dimensional optimization with two additional constraints. Two finite-dimensional approximations of the problem are defined. They are solved by two numerical algorithms which do not require calculation of the gradient of the payoff function. The first algorithm is an algorithm of simulated annealing with a uniform partition of the interval. The second algorithm uses a piecewise-constant approximation of the solution with a choice of the interval partition. Two illustrative examples connected with a resource allocation problem are considered. The numerical results are given and compared.

1. Introduction

Dynamic Stackelberg games (Basar and Olsder, 1999) are actively analyzed and discussed as adequate models of the hierarchically controlled dynamic systems. Thus, one of the interesting problem domains is resource allocation in organizational and economic systems (Christodoulou et al., 2015; Novikov, 2013).

Analytical methods of solution of the dynamic Stackelberg games are quite complicated due to the complex nature of those models. A comprehensive approach was proposed by Germeier for static Stackelberg games (Germeier, 1986) and developed by Kononenko and Gorelov for the dynamic case (Gorelov and Kononenko, 2015; Kononenko, 1977; Kononenko, 1980). The idea consists in the implementation of a cooperative trajectory and punishment in the case of defection.

However, the numerical algorithms are more convenient in this context. Evolutionary algorithms are especially useful, such as genetic and simulated annealing algorithms (Jones, 2008). An important place belongs to the methods which do not require the calculation of the gradient of the payoff function (Hazan, 2015).

The authors' approach is presented in (Belyavsky et al., 2016; Belyavsky et al., 2018a; Belyavsky et al., 2018b). In the paper Belyavsky et al., 2016 an application of the evolutionary modeling for the solution of the problems of sustainable management in active systems is considered. The different information structures of hierarchical differential games are described. The result which gives the opportunity of using of genetic algorithms for the solution of these problems is obtained and illustrated by a model example. In (Belyavsky et al., 2018a) a dynamic game theoretic model of resource allocation in the organizational system is proposed. The algorithms of evolutionary modeling are developed in this context and illustrated

[*] The research is supported by the Russian Science Foundation, project 17-19-01038.

by model examples. The paper (Belyavsky et al., 2018b) considers resource allocation among producers (agents) in the case where the Principal knows nothing about their cost functions while the agents have Markovian awareness about their strategies. We use a dynamic setup of the stochastic inverse Stackelberg game as the model and suggest an algorithm for solving this game based on Q-learning. The associated Bellman equations contain functions of one variable for the Principal and the agents.

This paper develops the described approach. In Section 2. the model formulation is given. Section 3. presents the Stackelberg game in infinite-dimensional and finite-dimensional spaces. In Sections 4. and 5. the simulated annealing and binary partition algorithms are exposed respectively. Section 6. is dedicated to the numerical results and their comparative analysis based on the first numerical example. The section 7. treats an application of the simulated annealing algorithm in a static game with incomplete information. The numerical results concerned with an additional illustrative example are given in Section 8.. Section 9. concludes.

2. A model formulation

A dynamic Stackelberg game with one leader and multiple followers (agents) is considered. The game theoretic model contains the following main elements: state of the game $(x_0, x(t)) \in R^2$, strategies $(u(t), v(t)) \in R^{r+1}$, leader's payoff $\int_0^T g_0(x_0, u, v)\, dt$, agents' payoffs $\int_0^T g_i(x, u, v)\, dt$; u — the leader's control; v_i — a reaction of the agent indexed by i.

Define the leader's problem as calculation of

$$\max_u \int_0^T g_0(x_0, u, v)\, dt, \quad \text{with constraint} \quad dx_0(t) = f_0(x_0, u)\, dt, \quad x_0(0) = x_0^0. \quad (1)$$

A homeostasis condition $x(t) \in X$ can be also added, for example, in the form $(x_0^* - x(t))^2 \leq a$. The homeostasis condition can be expressed by a penalty: $k \int_0^T (x_0^* - x(t))^2\, dt$. We can include the penalty into the leader's payoff functional and then consider a new payoff functional: $\int_0^T \left[g_0({}_0, u, v) - k(x_0^* - x(t))^2\right]\, dt$.

The agents' problems are set up in the form:

$$\max_{v_i} \int_0^T g_i(x, u, v)\, dt, \quad \text{with constraints} \quad dx_i(t) = f_i(x_i, v_i)\, dt, \quad x_i(0) = x_i^0. \quad (2)$$

It is supposed that the game (1), (2) can be transformed into a static Stackelberg game in the infinite-dimensional linear spaces:

$$J_0(u, v) \to \max_u; \quad J_i(u, v) \to \max_{v_i}, \quad i = 1, 2, \ldots, r. \quad (3)$$

In other words, for any feasible strategies (u, v) there is an algorithm of calculation of the state of the system (x_0, x_i) and the players' payoffs. Let us assume that the functions u and v_i belong to a Banach space $B[0,1]$ of the bounded functions with a uniform norm: $\|f\| = \sup_{t \in [0,1]} f(t)$. The normalization in time is made additionally. Thus, the game (3) is considered.

3. The Stackelberg game in infinite-dimensional and finite-dimensional spaces

The leader chooses her strategy u and reports it to the agents. In turn, the agents choose their strategies as a best response to the leader's strategy from the set of Nash equilibria in their game in normal form: $v(u) \in N(u)$. Therefore the leader's problem takes the form

$$\max_{u} \min_{v \in N(u)} J_0(u, v), \tag{4}$$

if the agents do not cooperate with her, and the form

$$\max_{u} \max_{v \in N(u)} J_0(u, v), \tag{5}$$

if they cooperate.

Let $\Phi(u)$ be the solution of the internal problem in (4) or (5). Then the leader's problem has the form

$$\max_{u} \bar{J}_0(u), \tag{6}$$

where $\bar{J}_0(u) = J_0(u, \Phi(u))$.

Consider a finite-dimensional approximation of the problem (6). A sufficient condition of the possibility of the finite-dimensional approximation is a continuity of the functional $J_0(u)$ on the set of feasible solutions. The continuity is ensured by the Lipshitz condition:

$|\bar{J}_0(u) - \bar{J}_0(w)| \leq L\|u - w\|$ which follows from the two inequalities:

$$\begin{gathered} |J_0(u_2, v) - J_0(u_1, w)| \leq L_u\|u_2 - u_2\| + L_v\|v - w\|_r, \\ \|\Phi(u_2) - \Phi(u_1)\|_r \leq L_\Phi\|u_2 - u_1\|. \end{gathered} \tag{7}$$

The last inequality in (7) is the most difficult for checking.

Consider the first class of feasible controls as a subset of the space of bounded functions in the form

$$\begin{aligned} L^1([0,1]) = \Big\{ u \in B[0,1] \colon \exists\, \alpha,\ \sup \frac{|u(t) - u(s)|}{|t - s|} \leq \alpha, \\ 0 \leq t \leq 1,\ 0 \leq s \leq 1,\ t \neq s \Big\}. \end{aligned} \tag{8}$$

It is assumed that the leader is in a sense restricted in her actions and therefore chooses her controls from this class. In other words, the leader is unable to make 'sharp motions'.

The next result forms a base for the proposed method of finite-dimensional approximation.

Theorem 1 (Belyavsky et al., 2016). *If $u \in L^1([0,1])$ then a sequence exists*

$$u^n(t) = u_0 + \alpha \sum_{i=1}^{n} \delta_i^n(u) I_{\{t > \tau_i\}}, \quad \delta_i^n(u) \in \{-1, 0, 1\},\ \tau_i = i/n,\ i = 0, 1, \ldots, n, \tag{9}$$

that converges to u by the norm of the space $B[0,1]$.

This result means that the subset

$$\bar{L}^1[0,1] = \Big\{u \in L[0,1] \colon \exists\,(u_0, \alpha, n, \delta),\ u = u_0 + \alpha \sum_{i=1}^{n} \delta_i I_{\{t>\tau_i\}},$$
$$\delta_i \in \{-1, 0, 1\},\ \tau_i = i/n,\ i = -0, 1, \dots, n\Big\}$$

is dense in $L_1[0,1]$.

Assume that the initial problem (6) with an additional constraint $u \in L_1[0,1]$ has a solution. Then the finite-dimensional approximation of the problem (6) with the additional constraint is the optimization problem

$$\max_{u \in \bar{L}^1[0,1]} \bar{J}_0(u) = \max_{u_0 \in R, \alpha, \varDelta} \bar{J}_0(u_0, \alpha, \varDelta), \quad \varDelta = (\delta_i)_{i=1}^{n}. \tag{10}$$

Notice that the problem (10) has a solution if for any fixed $\bar{\varDelta}$ the problem $\max\limits_{u_0 \in R, \alpha} \bar{J}_0(u_0, \alpha, \bar{\varDelta})$ has a solution. Suppose that the condition holds. The reason of the finite-dimensional approximation is given by the following result.

Theorem 2 (Belyavsky et al., 2016). *Let* $p^* = \max\limits_{u \in L[0,1]} \bar{J}_0(u) = \bar{J}_0(u^*)$*,* $q^* = \max\limits_{u \in \bar{L}[0,1]} \bar{J}_0(u) = \bar{J}_0(\bar{u}^*)$*. Then for any* $\epsilon > 0$ *we have* $p^* - q^* \le \epsilon$*.*

In fact, the continuity of the functional $J_0(u)$ and the density of the set $\bar{L}^1([0,1])$ in the set $L^1([0,1])$ imply the following inequalities: $p^* - q^* = J_0(u^*) - J_0(\bar{u}^*) \le J_0(U^*) - J_0(\bar{u}_1) \le \epsilon$. To satisfy the last inequality it is required to choose $\bar{u}_1 \in \bar{L}_1([0,1])$ close enough to u^*.

The second class of feasible controls is a set of bounded functions having on the segment $[0,1]$ a finite number of the points of discontinuity. Denote this set by $L^2([0,1])$. Define the set

$$\bar{L}^2([0,1]) = \Big\{u \in B[0,1] \colon \exists\,(n, (c_i)_{i=1}^{n}, (\tau_i)_{i=1}^{n}),\ 0 = \tau_0 < \tau_1 < \dots < \tau_n = 1,$$
$$f(t) = c_1 I_0(t) + \sum_{i=1}^{n} c_i I_{(\tau_{i-1}, \tau_i]}(t)\Big\}.$$

It is evident that the set $\bar{L}^2$ is dense in the set L^2. Given the continuity of the functional $\bar{J}_0(u)$ on the set $L^2([0,1])$ the problem

$$\max_{u \in \bar{L}^2[0,1]} \bar{J}_0(u) \tag{11}$$

is a finite-dimensional approximation of the problem (6) with the additional constraint $u \in L^2[0,1]$ if both problems have solutions.

4. A simulated annealing algorithm

The simulation annealing algorithm is used for the solution of the problem (10). Let us analyze the variables in the problem (10). The first two variables α and u_0 are real numbers, the third variable $\varDelta$ takes its values from the finite set of sequences $S = \{(\delta_i)_i^n \colon \delta_i \in \{-1, 0, 1\}\}$. If the functional $\bar{J}_0(u)$ satisfies a global Lipshitz condition with the Lipshitz constant L, and $K = \sup \alpha$ in the definition (8) then for a given ϵ the number of elements in the sequence is equal to $n = \left[\frac{KL}{\epsilon}\right] + 1$ (see details in Belyavsky et al., 2016). Thus, the considered problem is connected with calculation of the maximum of the function $F(\Delta) = \max\limits_{u_0,\alpha} \bar{J}_0(u_0, \alpha, \Delta)$ on the finite set S. Problems of that kind can be solved efficiently by the algorithms of evolutionary modeling, such as genetic algorithms and simulated annealing algorithms. The genetic algorithm was used in (Belyavsky et al., 2016; Belyavsky et al., 2018a), that's why here we consider the simulated annealing algorithm (Jones, 2008).

The algorithm starts from an initial Δ and an initial temperature $T = T_s$. The iterations have the following form.

1. The new $\bar{\Delta}$ is calculated in the neighborhood of the current Δ.
2. If $F(\bar{\Delta}) \geq F(\Delta)$ then $\Delta := \bar{\Delta}$ else $\Delta := \bar{\Delta}$ with probability

$$p = \exp\left(\frac{F(\bar{\Delta}) - F(\Delta)}{T}\right).$$

3. Set the temperature $T := qT$ where $0 < q < 1$.
4. The iterations are repeated until $(T \geq T_f) \wedge (|\Delta F| > \bar{\varepsilon})$.

5. A binary partition algorithm

Now suppose that the functional $\bar{J}_0(u)$ is additive, or for an arbitrary partition of the segment $[0, 1]$: $0 = \tau_0 < \tau_1 < \cdots < \tau_n = 1$ the inequality holds:

$$\bar{J}_0(u) = \bar{J}_0\Big(\sum_{i=1}^{n} u(t) I_{\{[\tau_{i-1},\tau_i)\}}(t)\Big) = \sum_{i=1}^{n} \bar{J}_0\Big(u(t) I_{\{[\tau_{i-1},\tau_i)\}}(t)\Big). \tag{12}$$

The algorithm has the following form.

1. Initialization. Let the initial partition of the segment is given: $[0, 1/2) \cup [1/2, 1]$. Consider the approximation $u^1(t) = c_{1,0} I_{[0,1/2]}(t) + c_{1,1} I_{(1/2,1]}(t)$. From all approximations in this form choose the best one using the property of additivity of the functional $\bar{J}_0(u^1) = \bar{J}_0(c_{1,0} I_{[0,1/2]}) + \bar{J}_0(c_{1,1} I_{(1/2,1]})$. For this purpose let's solve two independent problems: $\max\limits_{c} \bar{J}_0(c I_{[0,1/2]})$, $\max\limits_{c} \bar{J}_0(c I_{[0,1/2]})$.

2. Iterations. An iteration with the index n consists in the following. Let the current partition be $0 = \tau_0^{(n)} < \tau_1^{(n)} < \cdots < \tau_n^{(n)} = 1$, and the respective current approximation be $u^{(n)}(t) = c_{n,1} I_0(t) + \sum\limits_{i=1}^{n} c_{n,i} I_{(\tau_{i-1}^{(n)},\tau_i^{(n)}]}(t)$. Define the sequence $b_{n,j} = \frac{1}{\tau_j^{(n)} - \tau_{j-1}^{(n)}} \bar{J}_0\Big(c_{n,j} I_{\left(\tau_{j-1}^{(n)},\tau_j^{(n)}\right]}\Big)$ and the respective probabilities of choice of the interval: $p_{n,j} = \frac{b_{n,j}}{\sum\limits_{k=1}^{n} b_{n,k}}$, $j = 1, 2, \ldots, n$. The interval is chosen randomly according to the distribution of probabilities $p_{n,j}$. The chosen interval with the index j, namely $\left[\tau_{j-1}^{(n)}, \tau_j^{(n)}\right]$, is partitioned on two intervals of equal length $\left[\tau_{j-1}^{(n)}, \frac{\tau_{j-1}^{(n)} + \tau_j^{(n)}}{2}\right]$, $\left[\frac{\tau_{j-1}^{(n)} + \tau_j^{(n)}}{2}, \tau_j^{(n)}\right]$ and the new approximation

$$u^{n+1}(t) = c_{n+1,1} I_0 + \sum_{i=1}^{n+1} c_{n+1,i} I_{\left(\tau_{i-1}^{(n+1)},\tau_i^{(n+1)}\right]}(t)$$

is calculated, where $c_{n+1,i} = c_{n,i}$ if $i = 1, \ldots, j-1$;

$$c_{n+1,j} = \arg\max_c \bar{J}_0 \left(cI_{\left[\tau_{j-1}^{(n)}, \frac{\tau_{j-1}^{(n)}+\tau_j^{(n)}}{2}\right]} \right),$$

$$c_{n+1,j+1} = \arg\max_c \bar{J}_0 \left(cI_{\left[\frac{\tau_{j-1}^{(n)}+\tau_j^{(n)}}{2}, \tau_j^{(n)},\right]} \right);$$

$c_{n+1,i} = c_{n,i-1}$ if $i = j+2, \ldots, n+1$. The new partition $\tau_i^{(n+1)} = \tau_{i-1}^{(n)}$ is calculated if $i = 0, \ldots, j-1$, $\tau_j^{(n+1)} = \frac{\tau_{j-1}^{(n)}+\tau_j^{(n)}}{2}$, and $\tau_i^{(n+1)} = \tau_{i-1}^{(n)}$ if $i = j+1, \ldots, n+1$.

The iterations are repeated until the changes become small enough.

This algorithm is a monotonous one. Therefore the sequence $\bar{J}_0(u_n)$ converges if the leader's payoff functional is bounded from above. An objective of minimization of the number of points of the partition of the interval is aimed additionally.

6. The first example

We will consider a dynamic Stackelberg game which is already reduced to the form (6). The game describes a resource allocation between producers. Notice the monograph (Novikov, 2013) in this connection. An amount of the resource allocated in the moment t is denoted by $u(t)$. Each player receives his part of the resource $u_i(t)$; it is evident that $u(t) = \sum\limits_{i=1}^{r} u_i(t)$. Given the resource each player produces a good $v_i(t)$ so that to maximize his instant payoff. The leader tends to maximize the total production and uses the proportional distribution mechanism, or $u_i(t) = \gamma_i(t) v_i(t)$. Then $\sum\limits_{i=1}^{r} \gamma_i(t) v_i(t) = u(t)$. An instant payoff of the i-th follower is calculated as the difference between his part of the resource and production cost, or $P_i(t) = \gamma_i(t) v_i(t) - \varphi_i(v_i(t))$. The cost $\varphi_i(x)$ is a convex non-decreasing function defined on R^+, and $\varphi_i(0) = 0$. Thus, an auxiliary Stackelberg game arises in the following form:

$$\max_{\gamma \geq 0} \sum_{i=1}^{r} v_i \quad \text{with constraint} \quad \sum_{i=1}^{r} \gamma_i v_i = u, \tag{13}$$

$$\max_{v_i \geq 0} [\gamma_i v_i - \varphi_i(v_i)].$$

For simplicity from now on the time t is omitted. It is important to note that in this formulation the game between followers is decomposed by the independent optimization problems. If we assume that $\gamma_i = \gamma$ then the equality $\sum\limits_{i=1}^{r} \gamma_i v_i = u$ implies the equality $\gamma = \dfrac{u}{\sum_{i=1}^{r} v_i}$, and the solution of the followers' game consists in the calculation of the Nash equilibrium in the game with individual followers' problems: $\max \left[\dfrac{u v_i}{\sum_{j=1}^{r} v_j} - \varphi_i(v_i) \right]$ (see details in Christodoulou et al., 2015).

Consider the game with cost functions $\varphi_i(x) = \mu_i x^2$. In this case the solution of (11) takes the form

$$v_i = \frac{\gamma_i}{2\mu_i}, \quad \gamma_i = \sqrt{\frac{2u}{\sum_{j=1}^{r} \frac{1}{\mu_j}}}. \tag{14}$$

Notice that all γ_i do not depend on i.

The instant payoff of the leader is determined as $R\Big(\sum_i v_i(t), u(t)\Big)$. The function $R(x, y)$ increases in the first argument and decreases in the second argument. The total payoff is an integral of the instant payoff: $\displaystyle\int_0^T R\Big(\sum_{i=1}^{r} v_i, u\Big)\, dt$. Based on (14) we receive the total payoff as $\displaystyle\bar{J}_0(u) = \int_0^T R\left(\frac{1}{2}\sqrt{\frac{2u}{\sum_{j=1}^{r}\frac{1}{\mu_j}}}\sum\frac{1}{\mu_i}, u\right) dt$. Denote $\displaystyle A(t) = \sqrt{\frac{1}{2}\sum_{i=1}^{r}\frac{1}{\mu_i(t)}}$. Then $\displaystyle\bar{J}_0(u) = \int_0^T R(A\sqrt{u}, u)\, dt$. Assume that $F(x, y) = x - y$, then $\displaystyle\bar{J}_0(u) = \int_0^T [A\sqrt{u} - u]\, dt$. It is evident that the function $u^*(t) = \frac{A^2(t)}{4}$ is a maximizer for $\bar{J}_0(u)$. This function satisfies the first and second additional constraints if $\mu_i(t) > 0$.

Consider the algorithm of simulated annealing in the game with two followers and $\mu_1(t) = t^2 + 1$, $\mu_2(t) = 2t^2 + 1$. In this algorithm a partition of the interval is fixed. Calculate $F(\mathit{\Delta}) = \max\limits_{u_0,\alpha}\Big[\sum\limits_{i=1}^{n}(a_i\sqrt{u_0 + \alpha g_i(\mathit{\Delta})} - (u_0 + \alpha g_i(\mathit{\Delta}))\mathit{\Delta}t)\Big]$. From the concavity by u_0 and α follows that the optimal values are the solutions of the algebraic system of two equations:

$$\sum_{i=1}^{n}\frac{a_i}{\sqrt{u_0 + \alpha g_i(\mathit{\Delta})}} = 2, \quad \sum_{i=1}^{n}\left(\frac{a_i}{\sqrt{u_0 + \alpha g_i(\mathit{\Delta})}} - 2g_i(\mathit{\Delta})\mathit{\Delta}t\right) = 0.$$

In these formulas $\displaystyle a_i = \int_{\tau_{i-1}}^{\tau_i} A(t)\, dt$, $\displaystyle g_i(\mathit{\Delta}) = \frac{1}{n}\sum_{j=1}^{i-1}\delta_j$, $\mathit{\Delta}t = \tau_i - \tau_{i-1}$. For calculation $\tilde{\mathit{\Delta}}$ in the neighborhood of $\mathit{\Delta}$ each δ_i is replaced by $\tilde{\delta}_i \in \{-1, 0, 1\} \setminus \delta_i$ with probability q/n and equal probabilities on the set $\{-1, 0, 1\} \setminus \delta_i$. The parameter $q \in \{1, 2, \dots, n-1\}$ determines a mean number of the changing elements $\mathit{\Delta}$. The results of calculations are presented in Fig. 6..

Now consider an application of the binary partition algorithm to the same problem. For this purpose, the following problem is solved each time: for an interval $[\tau, s]$ it is required to find $\displaystyle\min_c\left[\sqrt{c}\int_\tau^s A(t)\, dt - c(s - t)\right]$. The optimal value is $\displaystyle c^* = \left(\frac{1}{2(s - \tau)}\int_\tau^s A(t)\, dt\right)$. The results of calculations are presented in Fig. 6..

7. A static game with incomplete information: The simulated annealing algorithm

Consider the following problem setup (Belyavsky et al., 2018b). The leader uses a resource allocation as an incentive for the agent and tries to $\max\limits_u[\psi(v) - \varphi(u, v)]$. The

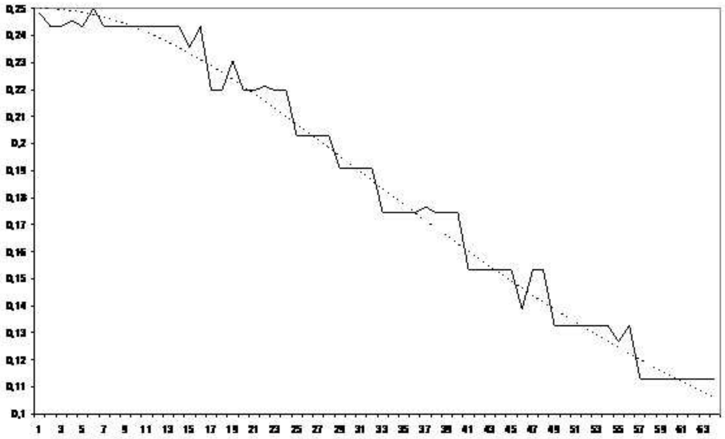

Fig. 1. The results of simulated annealing algorithm for $T_{\max} = 100$, $T_{\min} = 10$, $q = 5$, $n = 64$. The dotted line is the exact solution

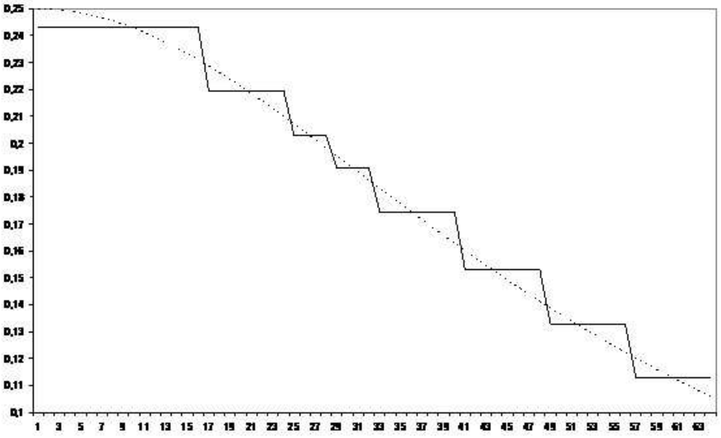

Fig. 2. The results of the binary partition algorithm. The dotted line is the exact solution

agent maximizes his profit: $\max_v [\varphi(u,v) - f(v)]$. Assume that a Nash equilibrium (u^*, v^*) exists in this game. The feature of the game is that the leader does not know the cost function f of the agent and cannot calculate the equilibrium respectively. So, the leader uses a sequence of controls $u(t)$ for the determination of u^*. The sequence $v(t)$, $t = 0, 1, \ldots$ represents the agent's best responses on $u(t)$. In each moment of time t the leader knows an interval $v(0), \ldots, v(t-1)$ of the best response sequence of the agent. Based on this information, the leader chooses the control $u(t) = u(t-1)(1 + \alpha\delta(t))$, according to which the agent receive the amount of resource $\varphi(v(t-1), u(t))$. Similarly to Section 3., $\delta(t) \in \{-1, 0, 1\}$, $0 < \alpha < 1$. Thus, in each iteration of the game the leader can save her control or increase/decrease it by a fixed value according to a 'Lipschitz' concept of the paper. The initial value $u(0)$ and the sequence $\delta(t)$ completely determine the sequence $u(t)$. Assume that the initial value $u(0) = x$ is known.

Consider the agent's problem. The agent supposes that $\delta(t)$ is a Markov sequence with the set of states $\{-1, 0, 1\}$ and a transition probabilities matrix Q with dimension (3×3). The initial probability distribution y on the set $\{-1, 0, 1\}$ is given.

The agent's problem is to calculate

$$\max E_{x,y} \sum_{t=1}^{\infty} \beta^t \big[\varphi(v(t-1), u(t)) - f(v(t))\big]. \tag{15}$$

If the function $-f(z) + \beta\varphi(z, w)$ is strictly concave for any value of the argument w then the optimal control of the agent is

$$v^*(x, y) = \arg\max_z \big[-f(z) + \beta\varphi\big(z, y(1 + \alpha(q_{x,3} - q_{x,1}))\big)\big] \tag{16}$$

(see details in Belyavsky et al., 2018b). Thus, the current reaction of the agent is calculated as

$$v(t) = \arg\max_z \Big[-f(z) + \beta\varphi\big(z, u(t)(1 + \alpha(q^t_{\delta(t),3} - q^t_{\delta(t),1}))\big)\Big]. \tag{17}$$

The matrix Q^t is calculated as a maximally likely estimation of the transition matrix for the interval $\delta(1), \ldots, \delta(t)$.

Now consider the leader's problem. The leader treats $v(t)$ as the best response to $\delta(t)$, and chooses the next value $\delta(t+1)$ so as the consequent agent's best response $v(t+1)$ is a random value. Thus, the leader solves the problem

$$\max_{\delta(t+1)} E\left[\psi\big(v(\delta(t+1))\big) - \varphi\big(v(t), u_t(1 + \delta(t+1))\big)\right]. \tag{18}$$

This problem is solved by the reinforcing learning algorithm. According to this algorithm, in each iteration the leader calculate the new value of Q-function:

$$Q_{t+1}(\delta(t)) = Q_t(\delta(t)) + h_t(R(\delta(t)) - Q_t(\delta(t))). \tag{19}$$

In (19) $R(\delta(t)) = \psi(v(t)) - \varphi(v(t-1), u_t)$, the initial value $Q_0(\cdot) \equiv 0$, and the sequence h satisfies the condition: $\sum_{t=1}^{\infty} h_t = \infty$, $\sum_{t=1}^{\infty} h_t^2 < \infty$. Then the distribution of probabilities on the set $\{-1, 0, 1\}$ is calculated as follows,

$$p_{t+1}(j) = \exp(Q_{t+1}(j)/T_{t+1}) \Big/ \sum_{k=-1}^{1} \exp(Q_{t+1}(k)/T_{t+1}), \quad j = -1, 0, 1; \tag{20}$$

and respectively $\delta(t+1)$ is chosen. In (20) T is a temperature that controls the degree of randomness in the choice of the next δ see Section 3.. The convergence of the algorithm is studied in (Sutton and Barto, 1998).

8. The second example

Consider another illustrative example where $\psi(x) = \sqrt{x}$, $\varphi(x,y) = xy$ и $f_i(x) = \mu x^2$. It is borrowed from (Belyavsky et al., 2018b) and slightly modified. The equality (17) takes the form $v(t) = \arg\max\limits_{z} \left[-\mu z^2 + \beta z u(t)\left(1 + \alpha(q^t_{\delta(t),3} - q^t_{\delta(t),1})\right)\right]$. A simple calculation gives $v(t) = \dfrac{\beta z u(t)\left(1 + \alpha(q^t_{\delta(t),3} - q^t_{\delta(t),1})\right)}{2\mu}$. The equilibrium solution in the game (15), (18) is $v^* = \dfrac{\beta u^*}{2\mu}$, $u^* = \left(\dfrac{\mu}{8\beta}\right)^{1/3}$. The equilibrium solution in the initial game is: $v^* = \dfrac{u^*}{2\mu}$, $u^* = \left(\dfrac{\mu}{8}\right)^{1/3}$. For $\mu = 1$, $\beta = 0.9$, $\alpha = 0.05$ the following numerical results are received:

Iteration	The leader's control	The leader's payoff
1	0.4	0
2	0.412	0.352264
3	0.39964	0.354196
4	0.411629	0.352203
5	0.423978	0.35414
6	0.436697	0.355904
7	0.436697	0.357482
8	0.449798	0.357482
9	0.463292	0.358856
10	0.449394	0.36001
11	0.462875	0.358817
12	0.476762	0.359978
13	0.491064	0.360902
14	0.491064	0.361569
15	0.491064	0.361569
16	0.476333	0.361569
17	0.490622	0.360877
18	0.505341	0.361553
19	0.520501	0.361952
20	0.504886	0.362054
21	0.520033	0.361944
22	0.504432	0.362055
23	0.504432	0.361936
24	0.504432	0.361936
25	0.504432	0.361936
26	0.504432	0.361936

Table 1. The numerical results for the Example 2

Note that the equilibrium solution of the leader for the given input data is equal to 0.519, and the respective payoff is 0.362.

9. Conclusion

The considered example with the known exact solution demonstrates a numerical applicability of both algorithms. As the algorithms require only the calculation of payoff functionals then they are efficient in situations when other methods do not generate implementable numerical schemes. It should be noticed that optimization methods which not require the calculation of gradient are actively discussed in the modern literature (see, for example, Hazan, 2015).

For a comparable precision of the approximate calculations given by the algorithms in the second example, these algorithms essentially differ. Their comparison by some important criteria is given in Table 2.

	A simulated annealing algorithm	A binary partition algorithm
Additional conditions on the leader's functional	absent	present (additivity)
Additional conditions on the leader's solution	present (Lipschitz codition)	practically absent
Pre-partition interval	require	do not require

Table 2. Comparison of algorithms

Note that in the static game with incomplete information (Section 7.) a binary partition algorithm is not applicable.

The comparison of the simulated annealing and the genetic algorithm results in the following conclusions. Both of them are random search algorithms in the optimization problems. An essential difference is that the simulated annealing is an algorithm of the depth search where only one potential solution is studied in each iteration, while the genetic algorithm is a width search algorithm which tests several potential solutions at each step. By the genetic algorithm a faster convergence may be expected, while the genetic algorithm has simpler iterations in a numerical sense. Therefore, the choice between them is determined by a specific problem. For example, if the calculation of the value of a finite-dimensional leader's objective function is a complicated problem, and the objective function is close to the concave one then the simulated annealing algorithm looks more attractive.

The considered methods are close to the method of scenarios used in simulation modeling.

References

Basar, T., Olsder, G. Y. (1999). Dynamic Non-Cooperative Game Theory. SIAM.

Belyavsky, G.I., Danilova, N.V.; Ougolnitsky, G.A. (2016). *Evolutionary modeling in sustainable management of active systems.* Math. Game Theory Appl., **8**(4), 14–29 (In Russian).

Belyavsky, G. I., Danilova, N. V., Ougolnitsky, G. A. (2018a). *Evolutionary methods for solving dynamic resource allocation problems.* Math. Game Theory Appl., **10(1)**, 5–22 (In Russian).

Belyavsky, G., Danilova, N., Ougolnitsky, G. (2018b). *A Markovian Mechanism of Proportional Resource Allocation in the Incentive Model as a Dynamic Stochastic Inverse Stackelberg Game.* Mathematics, **6(8)**, 131.

Christodoulou, G., Sqouritza, A., Tang, B. (2015). *On the Efficiency of the Proportional Allocation Mechanism for Divisible Resources.* M. Hoefer (Ed.): SAGT. LNCS 9347, 165–177.

Germeier, Yu. B. (1986). Non-antagonistic Games. Reidel Publishing Co., Dordrecht, Boston.

Gorelov, M. A., and Kononenko, A. F. (2015). *Dynamic models of conflicts. III. Hierarchical games.* Automation and Remote Control, **76(2)**, 264–277.

Hazan, E. (2015). *Introduction to Online Convex Optimization.* Foundations and Trends in Optimization, **2(3–4)**, 157–325.

Jones, M. T. (2008). Artificial Intelligence: A Systems Approach. Infinity Science Press, Hingham, MA.

Kononenko, A. F. (1977). *On multi-step conflicts with information exchange.* USSR Comp. Math. and Math. Phys., **17**, 104–113.

Kononenko, A. F. (1980). *The structure of the optimal strategy in controlled dynamic systems.* USSR Comp. Math. and Math. Phys., 13–24.

Novikov, D. (2013). Theory of Control in Organizations. Nova Science Publishers, New York.

Sutton, R. S., Barto, A. (1998). Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA.