Grid Method for Numerical Study of Time-Optimal Games with Lifeline *

Nataly V. Munts[0000−0003−3234−1267] and
Sergey S. Kumkov[0000−0002−2690−5380]

Krasovskii Institute of Mathematics and Mechanics, UrB RAS,
Yekaterinburg, Russia
natalymunts@gmail.com, sskumk@gmail.com

Abstract. The paper discusses a numerical grid method for solving time-optimal zero-sum differential games with lifeline. The dynamics of the considered games are supposed to be of a generic non-linear kind. The players' controls are taken from given compact sets of finite-dimensional Euclidean spaces. The objective of the first player is to reach the target set as fast as possible, with that, avoiding the set called lifeline. The second player counteracts to that: it tries either to guide the system to the lifeline avoiding the target set of the first player, or if it is impossible, to keep the system away from the target set infinitely, or if it is impossible too, to postpone maximally reaching the target set. In the text, we reference out work about theoretical constructions on existence of the value function of such a game. Also, we set forth the idea of the numerical method. Results of solving some model and practical examples are given.

Keywords: Time-optimal zero-sum differential games · Lifeline · Value function · Numeric grid method.

1 Formulation of Problem

The following autonomous dynamic system is considered:

$$\dot{x} = f(x, p, q), \qquad t \geq 0,\ x \in \mathbb{R}^d,\ p \in P,\ q \in Q. \tag{1}$$

Here, x is the d-dimensional state vector of the system, p and q are controls of the first and second players, respectively, which are constrained by compact sets in their finite-dimensional Euclidean spaces. Two sets are given: a compact set $\mathcal{T} \subset \mathbb{R}^d$ of the full dimension and an open set $\mathcal{W}$ such that $\mathcal{T} \subset \mathcal{W} \subset \mathbb{R}^d$. Denote $\mathcal{F} = \mathbb{R}^d \setminus \mathcal{W}$ and $\mathcal{G} = \mathcal{W} \setminus \mathcal{T}$. The set $\mathcal{T}$ is the target set. The first player tries to guide the system to it as soon as possible avoiding the set $\mathcal{F}$, which is called *lifeline*. The second player hinders this, it strives to reach the set $\mathcal{F}$ avoiding the set $\mathcal{T}$, of if it is impossible, to keep the system in the set $\mathcal{G}$ forever, or if this is impossible too, to postpone reaching the target set $\mathcal{T}$ as long as possible.

The following assumptions are supposed to be true:

C.1. the function $f : \mathbb{R}^d \times P \times Q \mapsto \mathbb{R}^d$ is continuous in the totality of variables and Lipschitzian on x with the constant λ; also, the Isaacs' condition [5] is held:

$$\min_{p\in P}\max_{q\in Q}\big\langle s, f(x,p,q)\big\rangle = \max_{q\in Q}\min_{p\in P}\big\langle s, f(x,p,q)\big\rangle =: \mathcal{H}(x,s) \quad \forall s\in\mathbb{R}^d. \quad (2)$$

C.2. the boundary $\partial\mathcal{G}$ (that is, the boundaries $\partial\mathcal{T}$ and $\partial\mathcal{F}$) is compact, twice smooth and has the curvature radius not less than some $r > 0$.

C.3. the boundary $\partial\mathcal{T}$ and the function f obey the following condition:

$$\min_{p\in P}\max_{q\in Q}\big\langle n_{\mathcal{T}}(x), f(x,p,q)\big\rangle < 0, \quad \forall x\in\partial\mathcal{T}.$$

C.4. the boundary $\partial\mathcal{F}$ and the function f obey the following condition:

$$\min_{p\in P}\max_{q\in Q}\big\langle n_{\mathcal{F}}(x), f(x,p,q)\big\rangle < 0, \quad \forall x\in\partial\mathcal{F}.$$

In two last conditions, $n_{\mathcal{T}}(x)$ and $n_{\mathcal{F}}(x)$ denote the unit outer normal to the boundary of the corresponding set at the point x, which belongs to the boundary. The sense of these conditions is the following: the first player has dynamic advantage near the boundary of the target set and can guide the system inside the set if the point is at the boundary. In the same way, the second player has the dynamic advantage near the lifeline.

For a given initial position x_0, these objectives of players can be formalized in the following way. Let $x(\cdot;x_0)$ be a trajectory of the system emanated from the point x_0 under some players' strategies. Define two instants

$$t_* = t_*\big(x(\cdot;x_0)\big) = \min\big\{t : x(t;x_0)\in\mathcal{T}\big\},$$
$$t^* = t^*\big(x(\cdot;x_0)\big) = \min\big\{t : x(t;x_0)\in\mathcal{F}\big\},$$

which are the first instants when the trajectory hits the sets $\mathcal{T}$ and $\mathcal{F}$, respectively. They equal $+\infty$ if the corresponding set is never hit by the trajectory. The payoff for the trajectory $x(\cdot;x_0)$ is defined as

$$\tau\big(x(\cdot;x_0)\big) = \begin{cases} +\infty, & \text{if } t_* = +\infty \text{ or } t^* < t_*, \\ t_*, & \text{otherwise.}\end{cases} \quad (3)$$

The players' strategies are feedback. To define a motion of the system under feedback strategies of the players, we use the formalization suggested by Krasovskii and Subbotin [6, 7].

In paper [8], the authors have proved the existence of the value function for games of this kind. The proof uses the positional ideology set forth in [6, 7].

In [8] together with the original game (1), a Dirichlet problem for the Hamilton–Jacobi PDE corresponding to the game is considered:

$$H\big(x, Du(x)\big) - u(x) = 0, \; x\in\mathcal{G}, \quad (4)$$
$$u(x) = 0 \text{ if } x\in\partial\mathcal{T}, \quad u(x) = 1 \text{ if } x\in\partial\mathcal{F}, \quad (5)$$

where $H(x,s) = \mathcal{H}(x,s) + 1$. The function $\mathcal{H}(x,s)$ is the Hamiltonian defined in (2).

Also, it is proved by the authors that problem (4), (5) under assumptions C.1–C.4 has a continuous generalized solution (in the viscous [3, 4] or minimax [10] sense), which coincides with the value function of game (1).

2 Idea of Numerical Method

First, note that the payoff (3) can have infinite values, what is inconvenient for numerical analysis. Let us consider a new payoff

$$J\big(x(\cdot;x_0)\big) = \begin{cases} 1 - \exp\big(-\tau\big(x(\cdot;x_0)\big)\big), & \text{if } \tau < +\infty, \\ 1, & \text{otherwise}, \end{cases} \tag{6}$$

which has its values in the interval $[0,1]$. This variable change is well-known as the *Kruzhkov's transform*. Denote by $v(x)$ the value function for this payoff. The suggested method constructs some approximations to the function $v(x)$.

For further numerical construction, we change the continuous time by a discrete one with instants 0, h, $2h$, $3h$, ..., and the continuous space by a grid $\mathcal{L} = \big\{(i_1k, i_2k, \dots, i_dk)\big\}$, $i_j \in \mathbb{Z}$. So, h and k are the steps of time and spatial discretization. The steps of spatial discretization along different axes can differ, but this does not affect the idea of the method. Below, a linear enumeration of the nodes of the grid is assumed: $\mathcal{L} = \{l_s\}_{s\in\mathbb{Z}}$.

Original trajectories $x(\cdot;x_0)$ of the system are changed by discrete ones:

$$x_n = x_{n-1} + hf(x_{n-1}, p_{n-1}, q_{n-1}), \quad n = 1,2,3,\dots,$$

where x_0 is the initial position, $p_n \in P$, and $q_n \in Q$.

The value function $w(l_s)$ of the discretized game can be characterized on the basis of the Dynamic Programming Principle:

$$\begin{cases} w(l_s) = \gamma \max\limits_{q\in Q} \min\limits_{p\in P} w_{loc}\big(l_s + hf(l_s,p,q)\big) + 1 - \gamma, & \text{if } l_s \in \mathcal{L}_{\mathcal{G}}, \\ w(l_s) = 0, & \text{if } l_s \in \mathcal{L}_{\mathcal{T}}, \\ w(l_s) = 1, & \text{if } l_s \in \mathcal{L}_{\mathcal{F}}. \end{cases}$$

Here, $\mathcal{L}_{\mathcal{T}}$, $\mathcal{L}_{\mathcal{G}}$, and $\mathcal{L}_{\mathcal{F}}$ are the subcollections of the nodes of the grid $\mathcal{L}$, which are located in the sets $\mathcal{T}$, $\mathcal{G}$, and $\mathcal{F}$, respectively. The coefficient $\gamma = e^{-h}$. The symbol w_{loc} denotes some local approximation of the function w between the nodes of the grid. The approximation can be piecewise-linear, polylinear, or some other. Each type of the approximation (or, at least, each class of approximations) needs its own proof of convergence of the method.

Let $\mathcal{M}$ be the set of infinite vectors with indices in $\mathbb{Z}$. For any infinite vector W, the operator $F : \mathcal{M} \to \mathbb{R}$ is element-wisely defined as follows:

$$F_s(W) = \begin{cases} \gamma \max\limits_{q\in Q} \min\limits_{p\in P} w_{loc}\big(z(l_s,p,q), W\big) + 1 - \gamma, & \text{if } l_s \in \mathcal{L}_{\mathcal{G}}, \\ 0, & \text{if } l_s \in \mathcal{L}_{\mathcal{T}}, \\ 1, & \text{if } l_s \in \mathcal{L}_{\mathcal{F}}. \end{cases}$$

Here, $z(l_s, p, q) = l_s + hf(l_s, p, q)$. We have proved that in the case when w_{loc} is the piecewise-linear or polylinear approximation between the grid nodes the operator F is a contraction map and its fixed point impose an approximation to the value function $v(x)$.

Note that actually, only values $w(l_s)$ for the nodes $l_s \in \mathcal{L}_\mathcal{G}$ are important. The values of w at other nodes are fixed and do not need to be stored and/or computed. So, if the set $\mathcal{G}$ is bounded, then it can be covered by some finite grid $\mathcal{L}_\mathcal{G}$, which can be represented in a computer.

So, values at these nodes can be set to some initial states and further repeatedly recomputed by the operator F. After such a recomputation, one gets converging sequences at the nodes, which after a sufficient number of iterations approximate well the ideal values $w(l_s)$. The latters in their turn by local approximation estimate well the value function v of game (1) with payoff (6) if the steps h and k are small enough.

In our work, we do not consider the questions about the rate of convergence of the algorithm. Hovewer, this algorithm is based on the numerical method from the work [1] where also the convergence rate theorem [1, Th. 3.4, pp. 140–144] is proved. We believe that in our case the rate of convergence is similar, but this fact has not been proved yet.

3 Examples

We have an own cross-platform realization of this numerical method written using the environment .NetCore 3.0 and language C# of version 6.0 or later. A single-threaded program was written and then, by means of the capabilities of C#, it was made multi-threaded in order to compute faster on multi-core processors. A processor used for computing examples is Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz with 6 cores and 12 threads.

Two following examples are connected with the classic time-optimal game "Homicidal chauffeur" originally suggested by R. Isaacs in his book [5]. A pursuing object (car with a bounded turn radius) tries to catch an evading one with the dynamics of simple motions (pedestrian). The original dynamics are

$$\dot{x}_p = w_1 \cos\psi, \quad \dot{y}_p = w_1 \sin\psi, \quad \dot{\psi} = \frac{w_1}{R} a,$$
$$\dot{x}_e = w_2 \cos b, \quad \dot{y}_e = w_2 \sin b.$$

Here, (x_p, y_p) and (x_e, y_e) are the geometric positions of the pursuer and the evader in the plane; ψ is the course angle of the car's velocity; w_1 is the magnitude of the linear velocity of the car; the value R/w_1 describes the minimal turn radius of the car. The control $a \in [-1, +1]$ of the pursuer shows how sharply the car turns: the value $a = -1$ corresponds to the maximally sharp right turn, the value $a = +1$ corresponds to the maximally sharp left turn, and $a = 0$ corresponds to the instantaneous rectilinear motion. The control $b \in [-\pi, \pi]$ of the pedestrian is the instantaneous direction of its velocity, which magnitude is w_2.

3.1 Example 1

A strong disadvantage of the original dynamics is their quite high dimension, namely, 5. However, the dynamics permit [5] a reduction of the dimension of the phase vector in the following way. Superpose the origin and the position of the pursuer. Direct the ordinate axis along the current vector of the pursuer's velocity. So, the new state position (x, y) of the system is two-dimensional and its dynamics are the following:

$$\dot{x} = -\frac{w_1}{R} ya + w_2 \sin b, \quad \dot{y} = \frac{w_1}{R} xa - w_1 + w_2 \cos b.$$

Here, x, y are the two-dimensional coordinates of a new object, which now is jointly controlled by the players. The first player (pursuer) tries to guide the system to the set $\mathcal{T} = \{(x, y) \in R^2 : (x-0.2)^2 + (y-0.3)^2 \leq 0.015^2\}$ keeping the trajectory inside the set $\mathcal{W} = [-1.5, 1.5] \times [-1, 1.5]$. The second one (pedestrian) hinder this.

The parameters of the game taken for a numerical experiment are $w_1 = 2$, $w_2 = 0.6$, $R = 0.2$. The example has been taken from [9]. The time step $h = 0.001$, the spatial step $k = 0.005$. The number of iterations equals 200. The total time of computation was 7 hours and 51 minutes.

The graph of the value function in the three-dimensional space x, y, $v(x, y)$ is given in Fig. 1.

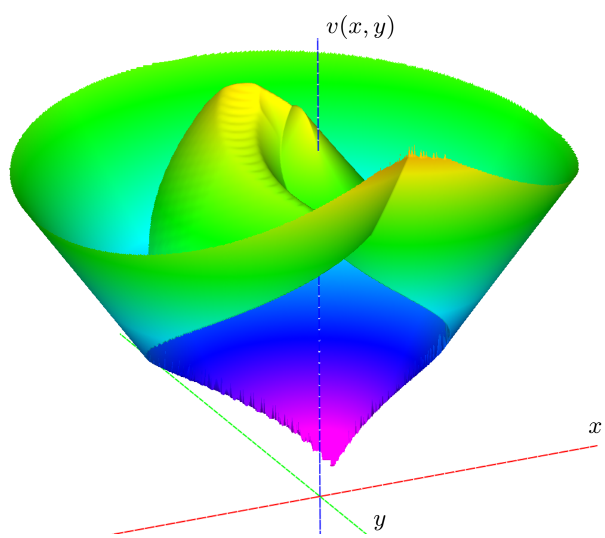

Fig. 1. The graph of the value function for Example 1.

3.2 Example 2

Now let us present a modified version of the problem having the following reduced dynamics:

$$\dot{x} = -\frac{Wy}{V_p}\sin\phi + V_e \sin\psi,$$
$$\dot{y} = \frac{Wx}{V_p}\sin\phi + V_e\cos\psi - V_p,$$
$$\dot{V}_p = W\cos\phi.$$

Here, x and y are the coordinates of the object; V_p is the current magnitude of the linear velocity of the car. Now, the pursuer manages two controls. The first one W is the magnitude of the acceleration of the car, which results in changing both the coordinates x, y. The second control is ϕ, which is the angle between the vectors of the acceleration and velocity of the car; it is assumed that $-\pi/2 \le \phi \le \pi/2$.

Note that due to chosen constraints for the control ϕ the velocity V_p can only grow. There are constraints for its magnitude: $V_p \in [V_{\min}, V_{\max}]$.

The value V_e is the magnitude of the velocity of the pedestrian; ψ is the angle between the velocity vector of the pedestrian and the direction of the y-axis ($0 \le \psi \le 2\pi$).

The target set is a cylinder $\mathcal{T} = \{(x, y, V_p) : x^2 + y^2 \le 0.3^2\}$. For computations, we take $W \equiv 1$, $V_e \equiv 0.3$, $V_{\min} = 0.5$, $V_{\max} = 1.5$. The example has been taken from [2].

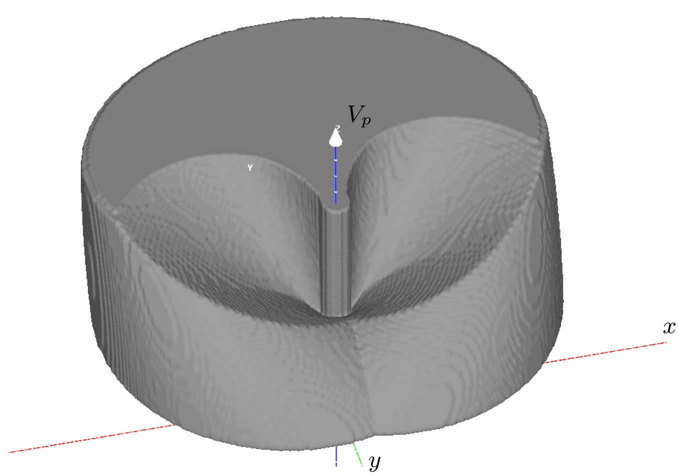

Fig. 2. A level set (Lebesgue set) of the value function for Example 2; the constant of the set equals 5.7.

So, the set $\mathcal{W} = [-6.0, 6.0] \times [-4.0, 7.0] \times [0.5, 1.5]$, time step $h = 0.05$, space steps $k = 0.05$, 0.05, 0.02, respectively. The number of iterations equals 150. The time of computation was 37 hours.

In this example the graph of the value function $v(x, y, V_p)$ is embedded into a four-dimensional space and cannot be drawn explicitly. Therefore, we show the level set (Lebesgue set) of the function, that is the collection of points (x, y, V_p) such that $v(x, y, V_p) \leq 5.7$ (see Fig. 2).

References

1. Bardi, M., Falcone, M., Soravia, P.: Numerical methods for pursuit-evasion games via viscosity solutions. In: Bardi, M., Parthasarathy, T., Raghavan, T.E.S. (eds.) Annals of the International Society of Dynamic Games, vol. 6: Stochastic and Differential Games, pp. 105–175. Birkhäuser, Boston (1999)
2. Botkin, N., Hoffmann, K.-H., Mayer, N., Turova, V.: Computation of value functions in nonlinear differential games with state constraints. In: Hömberg, D., Tröltzsch, F. (eds.) System Modeling and Optimization. CSMO 2011. IFIP Advances in Information and Communication Technology, vol. 391, pp. 235–244. Springer, Berlin, Heidelberg (2013)
3. Crandall, M.G., Evans, L.C., Lions, P.L.: Viscosity solutions of Hamilton-Jacobi equations. Transactions of the American Mathematical Society **277**(1), 1–42 (1983)
4. Crandall, M.G., Evans, L.C., Lions, P.L.: Some properties of viscosity solutions of Hamilton-Jacobi equations. Transactions of the American Mathematical Society **282**(2), 487–502 (1984)
5. Isaacs, R.: Differential games. John Wiley and Sons, New York (1965)
6. Krasovskii, N.N., Subbotin, A.I.: Positional differential games. Nauka, Moscow (1974) (in Russian)
7. Krasovskii, N.N., Subbotin, A.I.: Game-Theoretical Control Problems. Springer-Verlag, New York (1988)
8. Munts, N.V., Kumkov, S.S.: On time-optimal problems with lifeline. Dynamic Games and Applications **9**(3), 751–770 (2019)
9. Patsko, V.S., Turova, V.L.: Level sets of the value function in differential games with the homicidal chauffeur dynamics. International Game Theory Review **3**(1), 67–112 (2001)
10. Subbotin, A.I.: Generalized solutions of first order PDEs: the Dynamical optimization perspective. Birkhäuser, Boston (1995)