Proceedings of the
Eighth International Colloquium on Differential Equations

Plovdiv, Bulgaria, 18 - 23 August, 1997

***Editor:* D. Bainov**

///VSP///

UTRECHT, THE NETHERLANDS
TOKYO, JAPAN

PARALLEL ALGORITHM FOR CONSTRUCTION OF SINGULAR SURFACES IN LINEAR DIFFERENTIAL GAMES

S.S. Kumkov, V.S. Patsko

Institute of Mathematics & Mechanics, S.Kovalevskaya str., 16, Ekaterinburg, 620219, Russia
E-mail: kumkov.jr@ods.imm.intec.ru patsko@ods.imm.intec.ru

Abstract. The paper deals with an algorithm of construction and classification of singular surfaces in linear differential games with fixed terminal moment and convex payoff function which depends on two components of the phase vector. The algorithm is embedded into the backward procedures for building level sets of the value function. In this backward procedure, the level set section at the next moment is calculated using the section for the previous one. The data obtained during these calculations are used for detection and classification of the singular points in the new constructed section. Unrolled through the time, these singular points give the singular lines which go along over the surface of the level set. Singular lines are joined in the singular surfaces in the space of the game. The main idea of the parallel computations is briefly described.

Keywords: *Differential games, value function, singular surfaces, numerical methods, parallel computation algorithms.*

INTRODUCTION

As singular surfaces in the theory of differential games, such sets in the game space are named on which the optimal motions have some peculiarities (dispersion, refraction, junction, etc.). The classification of the singular surfaces was suggested in the R.Isaacs' book [1]. Necessary conditions which characterize this or that type of singularity were studied by P.Bernhard [2] and A.A.Melikyan [3]. In many papers (see, for example, [4, 5]), the analysis of the singular surfaces appearing in concrete differential games carried out.

In this paper, a computer algorithm for construction, analysis and classification of singular surfaces is shortly described.

We consider a linear antagonistic differential game [6]

$$\begin{gathered}\dot{x} = A(t)x + B(t)u + C(t)v, \\ x \in R^n, \quad u \in P, \quad v \in Q, \quad \varphi(x_i(T), x_j(T))\end{gathered} \tag{1}$$

with fixed terminal time T and convex payoff function φ which depends on two coordinates x_i, x_j of the phase vector. The first (second) player governs the control u (v) choosing that from the convex compact P (Q) and minimizes (maximizes) the value of the function φ at the moment T.

It is known that the substitution $y(t) = X_{i,j}(T,t)x(t)$, where $X_{i,j}(T,t)$ is a matrix combined of two rows of the fundamental Cauchy matrix, provides the transformation to the equivalent differential game of the second order.

For the linear differential game (1) at the beginning of the 80's, in the Institute of Mathematics and Mechanics, Ekaterinburg, there were elaborated [7–11] the backward constructions for building level sets of the value function.

Described in this paper algorithm for constructing the singular surfaces is embedded into the algorithm [8] for building level sets of the value function. The singular surfaces in the three-dimensional space t, y_1, y_2 of the equivalent game are combined of the singular lines. In its turn, the singular lines are built of the singular points which are selected on the sections of the concrete level set during the process of constructing such sections. If the original game is of the second order on its phase variable, then the reverse transition to the original phase coordinates t, x_1, x_2 is possible.

Till now, the algorithm has been elaborated only for the case of scalar controls of the first and second players. Hence, calculating the singular surfaces, it is supposed that the sets P and Q are segments.

The complex of programs was elaborated which permits to build the collection of level sets of the value function onto the given grid of its values. In the case of scalar players' controls, it additionally permits to construct and classify the singular surfaces. There exists the special version of these programs intended for the parallel multiprocessor computers.

1. CONSTRUCTING LEVEL SETS OF THE VALUE FUNCTION

Below, the algorithm from the paper [8] for constructing level sets of the value function is considered.

1.1. Backward procedure

Assume that the transfer from the game (1) with the payoff function φ depending on 2 coordinates of the phase vector to the equivalent game

$$\dot{y} = D(t)u + E(t)v, \qquad y \in R^2, \quad u \in P, \quad v \in Q, \quad \varphi(y(T)) \tag{2}$$
$$D(t) = X_{i,j}(T,t)B(t), \quad E(t) = X_{i,j}(T,t)C(t),$$

is already done. Let on the interval $[0,T]$ the sequence of moments $t_i: \ t_N = T, \ldots, t_i = t_{i+1} - \Delta, \ldots, t_0 = 0$ dividing the interval with the step Δ is given. The interest is in finding the level sets $W_c(t_i) = \{y \in R^2 : V(t_i, y) \leq c\}$ of the value function V for the given value of parameter c.

Replace the dynamics (2) by the piecewise-constant dynamics

$$\dot{y} = \mathbf{D}(t)u + \mathbf{E}(t)v, \quad \mathbf{D}(t) = D(t_i), \quad \mathbf{E}(t) = E(t_i), \quad t \in [t_i, t_{i+1}). \tag{3}$$

Instead of the sets P and Q, let us consider their polyhedral approximations $\mathbf{P}$, $\mathbf{Q}$. Let $\hat{\varphi}$ be the approximating payoff function. For any c, its level set $\mathbf{M}_c = \{y : \hat{\varphi}(y) \leq c\}$ is a convex polygon.

The approximating game (3) is taken so that, for each step $[t_i, t_{i+1}]$ of the backward procedure, we deal with the game with simple motions, polyhedral convex control constraints and the convex polyhedral target set. Put $\mathbf{W}_c(t_N) = \mathbf{M}_c$, next, find the game solvability set $\mathbf{W}_c(t_{N-1})$, then $\mathbf{W}_c(t_{N-2})$, and so on. As a result, the collection of convex sets is obtained which approximate in the Hausdorff metrics the level sets $W_c(t_i)$ of the value function in the game (2).

Let $\mathcal{P}(t_i) = -\Delta D(t_i)\mathbf{P}$, $\mathcal{Q}(t_i) = \Delta E(t_i)\mathbf{Q}$. The support function $l \to \rho(l, \mathbf{W}_c(t_i))$ of the set $\mathbf{W}_c(t_i)$ is the convex hull [12] of the function

$$\gamma(l, t_i) = \rho(l, \mathbf{W}_c(t_{i+1})) + \rho(l, \mathcal{P}(t_i)) - \rho(l, \mathcal{Q}(t_i)).$$

The function $\gamma(\cdot, t_i)$ is positively-homogeneous and piecewise-linear. The property of local convexity of this function can be violated only at the frontier of the linearity cones of the function $\rho(\cdot, \mathcal{Q}(t_i))$, i.e. at the frontier of the cones generated by the normals to the edges of the polygon $\mathcal{Q}(t_i)$ which have the common vertex.

1.2. Algorithm of convex hull construction

Let us agree to omit the argument t_i in the notation of the function γ. The linearity cones of γ are determined by the normals to the convex polygons $\mathbf{W}_c(t_i)$, $\mathcal{P}(t_i)$, $\mathcal{Q}(t_i)$. Gathering the outer normals of these sets and ordering them clockwise, the collection L of the vectors is obtained. The collection of values $\gamma(l)$ of the function γ on the vectors $l \in L$ is denoted by Φ. The collections L, Φ describe completely the function γ.

The collection of the normals to $\mathcal{Q}(t_i)$ ordered clockwise is denoted by S. The collection S is called the collection of "suspicious" vectors. This name is connected with the fact that the function γ is locally convex on the cones which interior does not contain the normals of the set $\mathcal{Q}(t_i)$. The violation of the local convexity can appear only on the cones which interior contains at least one normal of the polygon $\mathcal{Q}(t_i)$.

Let $L^{(1)} = L$, $\Phi^{(1)} = \Phi$, $S^{(1)} = S$. The $k+1$ step of the multistep convexing process consists in replacing the collections $L^{(k)}$, $\Phi^{(k)}$ by the collections $L^{(k+1)} \subset L^{(k)}$, $\Phi^{(k)} \subset \Phi^{(k+1)}$. The collection $S^{(k)}$ is also replaced by the new one $S^{(k+1)}$.

Describe now one step of the convexing process. Suppose that the angle between two neighboring vectors from the collection $L^{(k)}$ counted clockwise is less than π. Let $l \to \gamma^{(k)}(l)$ be the piecewise-linear function determined by the collections $L^{(k)}$, $\Phi^{(k)}$. Since $L^{(k)} \subset L^{(k-1)} \subset \ldots \subset L^{(1)}$, $\Phi^{(k)} \subset \Phi^{(k-1)} \subset \ldots \subset \Phi^{(1)}$, then for any vector $\bar{l} \in L^{(k)}$ the value $\gamma^{(k)}(\bar{l})$ is equal to $\gamma(\bar{l})$.

Take some vector $l_* \in S^{(k)}$ and check the local convexity of the function $\gamma^{(k)}$ on the cone generated by the vector l_* and two its neighboring vectors l_- and l_+ selected counterclockwise and clockwise from the collection $L^{(k)}$. In other words, check whether the inequality $l'_* y \le \gamma(l_*)$ is active in the triple of the inequalities $l'_- y \le \gamma(l_-)$, $\quad l'_* y \le \gamma(l_*)$, $\quad l'_+ y \le \gamma(l_+)$. If the system of three inequalities is compatible, then (by virtue of the ordering the vectors l_-, l_*, l_+) only the middle one can be inactive.

The algorithm of verification: find the intersection point y_* of the straight lines $l'_- y = \gamma(l_-)$ and $l'_* y = \gamma(l_*)$, and then check the inequality $l'_+ y_* < \gamma(l_+)$. If it holds, the local convexity takes place (the middle inequality is active). In the opposite case, the local convexity is absent (the middle inequality is inactive).

In the first case, the vector l_* is taken away from the collection $S^{(k)}$, and the remained set is denoted by $S^{(k+1)}$. Let $L^{(k+1)} = L^{(k)}$, $\Phi^{(k+1)} = \Phi^{(k)}$.

In the second case, two situations are distinguished. Let α be the angle counted clockwise from l_- to l_+:

- $\alpha < \pi$. The vector l_* is taken away from the collection $S^{(k)}$, and, simultaneously, the vectors l_- and l_+ are included into the collection (one of them or even both could be already presented in the collection $S^{(k)}$). Denote the new collection of the "suspicious" vectors by $S^{(k+1)}$. The difference of the new collection $L^{(k+1)}$ from the collection $L^{(k)}$

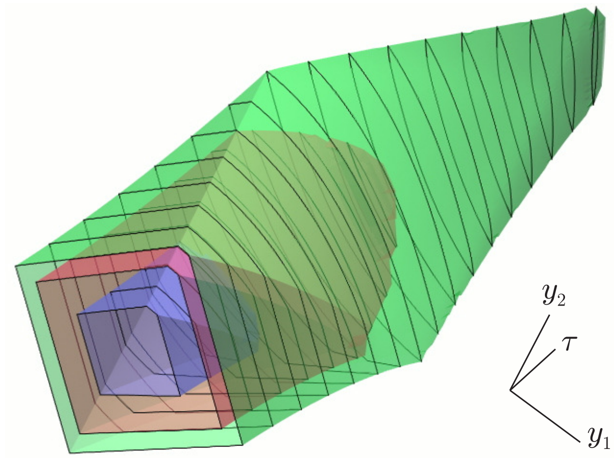

Figure 1: Three level sets of value function for game (4).

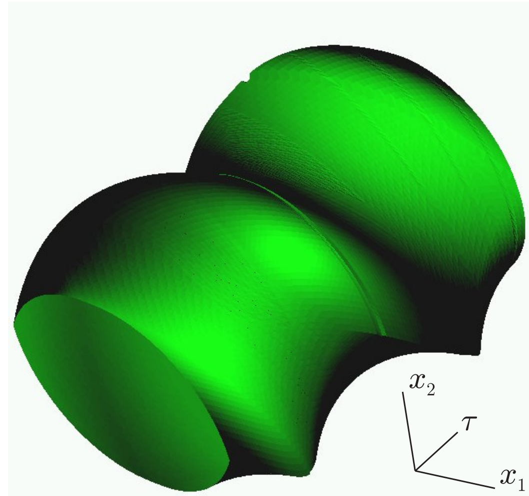

Figure 2: One level set of value function for game (5).

is that the vector l_* is absent in $L^{(k+1)}$. When processing $\Phi^{(k)}$ to $\Phi^{(k+1)}$, the values $\gamma^{(k)}(l_*) = \gamma(l_*)$ are taken away;

• $\alpha \geq \pi$. It means that the discussed triple of inequalities is incompatible. Thus, the convex hull of the function γ does not exist, i.e. $\mathbf{W}_c(t_i) = \emptyset$. The constructing is ceased.

One step of the convexing algorithm has been described. The algorithm finishes at the step with the number j, when for the first time $S^{(j)} = \emptyset$, i.e. when the collection of the "suspicious" vectors is empty. It means that the function $\gamma^{(j)}$, which corresponds to the collections $L^{(j)}$ and $\Phi^{(j)}$, is locally convex everywhere. Thus, the function $\gamma^{(j)}$ is the convex hull of the function γ. The second variant of termination is the following: the angle α between the vectors l_- and l_+ becomes greater or equal to π after rejecting the checked vector l_* from the collection of the "suspicious" vectors at some step. It means that $\mathbf{W}_c(t_i) = \emptyset$.

In Figure 1, three level sets of the value function are shown for the differential game

$$\begin{aligned} \dot{x}_1 &= x_2 + v \\ \dot{x}_2 &= u \end{aligned} \qquad |u| \leq 1,\ |v| \leq 1, \quad \varphi(x_1, x_2) = |x_1| + |x_2|. \tag{4}$$

The picture is represented in the equivalent coordinates. The symbol $\tau = T - t$ denotes the backward time. All three sets have terminated.

In Figure 2, the level set of the value function is shown for the differential game

$$\begin{aligned} \dot{x}_1 &= x_2 + v \\ \dot{x}_2 &= -x_1 + u \end{aligned} \qquad |u| \leq 1,\ |v| \leq 0.9, \quad \varphi(x_1, x_2) = x_1^2 + x_2^2. \tag{5}$$

The picture is given in the original coordinates.

2. CONSTRUCTION OF SINGULAR SURFACES

In this section, it is supposed that the controls of both players are scalar. So, the sets $\mathcal{P}(t_i)$ $\mathcal{Q}(t_i)$ are segments in the space y_1, y_2.

On each time interval $[t_i, t_{i+1})$, we have the game with simple motions [1] as the approximating game. The first player tries to pass the system from the polygon $\mathbf{W}_c(t_i)$ onto

Table of singularities classification

Number of normals	Flags	Additional condition	Object for marking	Secondary condition	Type of singularity
2	FS , FS	$\mathcal{Q}$-normal is strictly between	Vertex	$\mathcal{P}$-normal is not between	Dispersal for 2nd
				$\mathcal{P}$-normal is between	Dispersal for both
2	FS , $N\mathcal{P}$+FS	$\mathcal{Q}$-normal is strictly between	$\mathcal{P}$-normal edge		Equivocal
1	$N\mathcal{P}$		Edge	Normal is not equal to W-normal	Switching for 1st
				Normal is equal to W-normal	"Fuzzy" switching for 1st
1	FS	Normal is equal to $\mathcal{Q}$-normal	Edge		"Fuzzy" dispersal for 2nd
1	$N\mathcal{P}$+FS	Normal is equal to $\mathcal{Q}$-normal	Edge	Normal is not equal to W-normal	Switching for both
				Normal is equal to W-normal	"Fuzzy" switching for both

the polygon $\mathbf{W}_c(t_{i+1})$, the second player hinders to prevent it. Entering the discrimination of the second player, determine the optimal motions.

Let us fix some arbitrary point y_0 on the boundary of the polygon $\mathbf{W}_c(t_i)$. The second player's control constant on the interval $[t_i, t_{i+1})$ is called the optimal one if the first player can not direct the motion from the point y_0 inside the polygon $\mathbf{W}_c(t_{i+1})$. For fixed optimal control of the second player, the first player's control (also constant on the same interval) is called the optimal one if the motion gets onto the boundary of the set $\mathbf{W}_c(t_{i+1})$. The system motion generated by the optimal players' controls is called the optimal.

The point y_0 is called the regular if the optimal motion getting out from this point is unique and generated by the extreme players' controls. A point, which is not regular, is called the singular one.

Below, the classification of the singular points of the polygon $\mathbf{W}_c(t_i)$ is described. It is based on the analysis of the character of the optimal motions getting out from these points. For the classification, we shall use two marks. These marks attach to normals participated in the process of convex hull construction of the function $\gamma(\cdot, t_i)$. The mark FS ("former suspicious") is added to such normals which, during the convexing process, were denoted as "suspicious" ones, but, after the process, were remained in the collection which determines the polygon $\mathbf{W}_c(t_i)$. The mark $N\mathcal{P}$ is added to the normals which were taken from the set $\mathcal{P}(t_i)$. The classification is represented in Table.

Table deals with the normals from the collection obtained after the process of convexing. Individual normals or pairs of neighbor ones are analyzed. The number in the first column describes the number of normals considered in this or that combination of normals. In the second column, the marks are represented whose presence is checked out on the considered normals. With that, if only one mark is shown for some concrete vector, then the second mark is considered to be absent. The third column contains the additional conditions. Marks in the second column, together with the satisfaction of the additional condition, determine in whole the singularity type of the object from the forth column. Such object can be the polygon $\mathbf{W}_c(t_i)$ vertex which is incident for edges determined by normals (row 1), either the edge correspondent to the considered normal (rows 3–5) or one of two shown normals (row 2). Inside some cases, there exists subdivision which is determined by the condition from the fifth column. In the last column, the type of singularity is indicated which is finally prescribed to the mentioned object.

The term "$\mathcal{P}$-normal" ("$\mathcal{Q}$-normal", "W-normal") means a normal taken from the set $\mathcal{P}(t_i)$ ($\mathcal{Q}(t_i)$, $\mathbf{W}_c(t_{i+1})$).

Note, that length of the singular edge is in the linear proportion to the time step Δ if the normal to such edge was absent in the normals collection which determined the polygon $\mathbf{W}_c(t_{i+1})$.

Denominations of the singular surfaces are determined by a character of the optimal motions getting out from the marked point. The names are coordinated with ones used in the R.Isaacs' book.

The term "dispersal" corresponds to the situation when the bundle of the optimal motions getting out from the given point contains two motions generated by different extreme controls of the second player and one or two extreme controls of the first player.

The term "equivocal" corresponds to the situation when, from the vertex determined by the normals with marks $N\mathcal{P}{+}FS$ and FS, two optimal motions go out and those are generated by different extreme controls of the second player. Herewith, the first player's control for one of these motion is of extreme value, but is not such for the other motion. The latter motion comes to such vertex of the polygon $\mathbf{W}_c(t_{i+1})$ in which the cone of normals contains the considered normal with the mark $N\mathcal{P}{+}FS$. All other points of the marked edge are routine, namely, each point generates only one motion which comes to the mentioned vertex.

It is a very complicated case, formally included in Table into the "equivocal" class, when the normal marked as $N\mathcal{P}{+}FS$ is equal to a W-normal. Maybe, this case have to be considered and classified separately. However, it appears quite rarely and therefore the "equivocal" row in Table is not divided.

The term "switching" corresponds to the situation when the bundle of the optimal motions from each internal point of the singular edge contains a motion generated by one extreme control of the second player and non-extreme control of the first player.

On different sides of the equivocal edge, both players' controls change. In dispersal and switching situations, only one of players can change its control or they can change their controls together. In these two latter situations, a special subdivision appears. With that, it is indicated which players change their controls: the first, the second or both.

The term "fuzzy" means the special situation when two or three normals coincide. For example, the normals from the sets $\mathcal{Q}(t_i)$ and $\mathbf{W}_c(t_{i+1})$ are equal. Herewith, the infinite non-uniqueness of the optimal controls appears, and the bundle of the optimal motions diffuses and becomes "fuzzy". Such case is not one of the general position.

It can be shown that any point which is neither marked vertex nor included to any marked edge is the regular one. And vice versa, any point which is marked vertex or included to a marked edge is the singular, except, maybe, the case when it is an endpoint of a marked edge.

The data for the algorithm of singularities classification are accumulated during the process of the convex hull constructing, and after, the algorithm of classification begins to

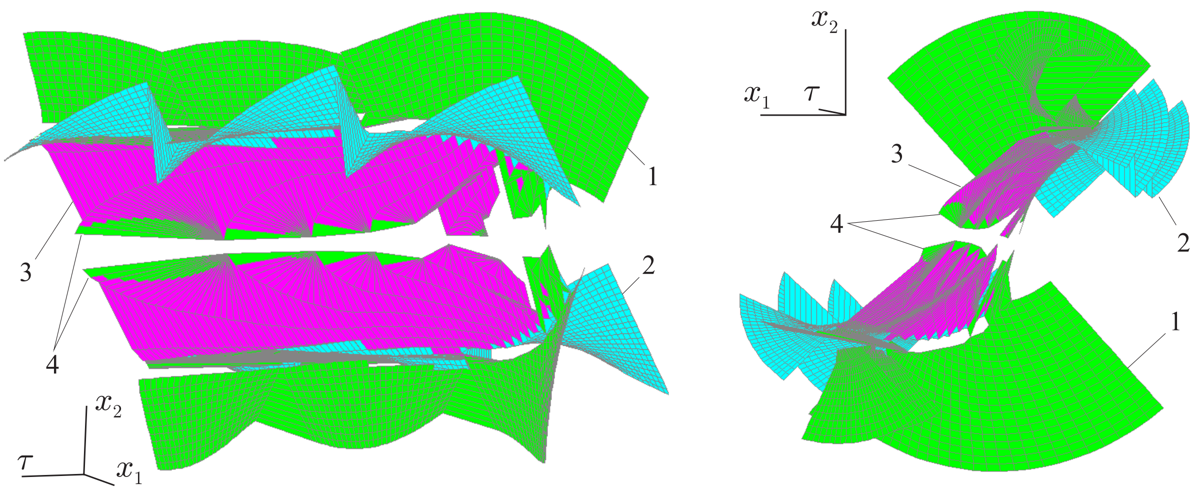

Figure 3: Singular surfaces for game (6). *Left*: View from the axis x_1. *Right*: View from the axis τ. *Notations*: 1 – dispersal surface for the second player, 2 – switching surface for the first player, 3 – equivocal surface, 4 – dispersal surface for both players.

work. Really, its work is reduced to the verification of the presence of situations described by rows of Table. As the result, the collection of the points and intervals with description of type of their singularity is obtained for the next backward time section of the level set.

For graphical presentation of the singular surfaces, a collection of the level sets is calculated on the given grid of magnitudes of the value function, and, simultaneously, the singular points and edges are detected on each section of each level set. Constructing the singular surfaces on the base of the singular points and intervals is carried out not in this program, but later, in the program of visualization.

The validity of the elaborated algorithm was verified on the test example with the dynamics from the game (4) but with different payoff function $\varphi(x_1, x_2) = |x_1|,\ x_2 = 0$; $\varphi(x_1, x_2) = +\infty,\ x_2 \neq 0$. The singular surfaces appeared in this example had been investigated in details in [13]. Results of our calculations coincide well with the results of the mentioned paper.

Figure 3 shows the picture of the singular surfaces for the differential game

$$\begin{aligned} \dot{x}_1 &= x_2 + v \\ \dot{x}_2 &= -x_1 + u \end{aligned} \qquad |u| \leq 1,\ |v| \leq 0.9, \quad \varphi(x_1, x_2) = \max\{|x_1|, |x_2|\}. \tag{6}$$

The surface close to the axis τ is dispersal for both players. On some distance from the axis τ, the equivocal surface situated. It comes into the switching surface for the first player and dispersal one for the second. The "empty" parts round the time axis visible in Figure 3 can be filled by calculations with smaller time step Δ of the backward procedure.

Calculations of many examples show that the presence of the dispersal for both players surface passing through the time axis and the presence of the equivocal surface on some distance from the axis are typical for the differential games of the second order with the dynamics and payoff function symmetrical relatively zero.

3. ORGANIZATION OF PARALLEL COMPUTATIONS

Parallelizing of computations was fulfilled for the computer of the *Paragon* type. It includes the collection of the i860 Intel processors with shared memory. Information is transferred to a customer through the host IBM-compatible computer included into the local area network. The access is also possible via the Internet.

In the discussed problem, the level sets of the value function and singular lines on them are built independently for different values of the function. So, the parallelizing of computations is natural, and computation and building each separate level set are also independent processes and are carried out on a separate processors. In the case of many sets calculations, some processors can calculate several sets sequentially. Different level sets can demand different computer time. So, the effective procedure of distribution of calculation tasks for each separate processor is the central point in the parallel algorithm.

The parallel algorithm for constructing level sets of the value function is described in [14]. Now, it has been modified for implementing the additional calculations on detection and classification of the singular surfaces. The algorithm of the singular surfaces building embeds well into the algorithm of the level sets calculations. So, the modification is joined only with including the block of the singular points classification into the main calculation procedure. With that, all governing procedures including the task manager are hold.

ACKNOWLEDGMENT

This research was supported by the Russian Foundation of Basic Researches under Grant No. 97-01-00458.

REFERENCES

[1] R.Isaacs, Differential games, John Wiley and Sons, New York (1965).

[2] P.Bernhard, in: Lecture Notes in Control and Information Sciences, 3, P.Hagedorn, H.W.Knobloch and G.J.Olsder, (Eds.), Springer-Verlag, Berlin (1977) pp. 1–33.

[3] A.A.Melikyan, in: Lecture Notes in Control and Information Sciences, 156, R.P.Hämäläinen and H.K.Ehtamo, (Eds.), Springer-Verlag, Berlin (1991) pp. 81–90.

[4] A.W.Merz, The homicidal chauffeur – a differential game, PhD dissertation, Standford University (1971).

[5] J.Shinar, M.Medinah and M.Biton, JOTA, 43(3), 431–456 (1984).

[6] N.N.Krasovskii and A.I.Subbotin, Game-theoretical control problems, Springer-Verlag, New York (1988).

[7] A.I.Subbotin and V.S.Patsko (Eds.), Algorithms and programs for solving linear differential games, Inst. of Math. and Mech., Sverdlovsk (1984) (in Russian).

[8] E.A.Isakova, G.V.Logunova, and V.S.Patsko, in: ref. [7] (1984) pp. 125–158 (in Russian).

[9] N.D.Botkin, in: ref. [7] (1984) pp. 5–38 (in Russian).

[10] A.M.Taras'ev and V.N.Ushakov, in: ref. [7] (1984) pp. 159–190 (in Russian).

[11] A.M.Taras'ev and V.N.Ushakov, in: Investigations of minimax control problems, A.I.Subbotin and V.S.Patsko, (Eds.), Inst. of Math. and Mech., Sverdlovsk (1985), pp 82–90 (in Russian).

[12] B.N.Pschenichnyi and M.I.Sagaidak, Cybernetics, 6(2), 72–83 (1970).

[13] V.S.Patsko and S.I.Tarasova, Engrg. Cybernetics, 22(5), 59–67 (1985).

[14] S.S.Kumkov, in: Developing and application of high performance computational facilities and programs on the base of parallel technologies in IMM UrO RAS, Inst. Math. and Mech., Report, Ekaterinburg (1996) pp. 114–121 (in Russian).