

## ЗАДАЧА ПРЕСЛЕДОВАНИЯ ОБЪЕКТОВ, ПЕРЕДВИГАЮЩИХСЯ ПО РАЗНЫМ ПОВЕРХНОСТЯМ

А. А. Дубанов

Бурятский государственный университет, г. Улан-Удэ  
alandubanov@mail.ru

В настоящей статье предлагается описание математической модели одной из задач преследования, когда преследуемый объект движется по поверхности, заданной точечным базисом, а преследующий объект движется по другой поверхности, расположенной над исходной. Скорости объектов в нашей задаче являются постоянными по модулю величинами. «Инертность» объектов моделируется при помощи угловой скорости вращения. Моделирование процесса движения выполняется в системе компьютерной алгебры MathCAD. В данной статье приведено описание вычислительного алгоритма, по которому в системе MathCAD была разработана программа, реализующая в режиме реального времени процесс преследования и убегания (уклонения).

**Ключевые слова:** задача преследования, объект преследования, преследующий объект, динамический базис, локальный базис, смена координат.

## PURSUIT PROBLEM OF OBJECTS MOVING ALONG DIFFERENT SURFACES

А. А. Дубанов

Buryat State University named after D. Banzarov, Ulan-Ude  
alandubanov@mail.ru

This article considers a description of the mathematical model of one of the pursuit problems when the pursued object moves along a surface defined by a point basis, and the pursuing object moves along another surface located above the original surface. The speeds of the objects in our problem are constant modulo values. The “inertness” of objects is modeled using an angular rate of rotation. The simulation of the movement process is performed in the computer system MathCAD. This article describes the computational algorithm that is used to develop a program in the MathCAD system that implements the process of pursuing and evading the problem in real time.

**Keywords:** pursuit problem, pursuit object, pursuing object, dynamic basis, local basis, change of coordinates.

**Введение. Постановка задачи.** В данной статье рассматривается задача преследования, когда преследуемый объект («Кролик») передвигается по поверхности  $\vec{P}_{rabbit}(u, v)$ , а преследующий объект движется по поверхности  $\vec{P}_{fox}(u, v)$ . В точках  $\vec{R}_{rabbit}(T)$  и  $\vec{R}_{fox}(T)$  поверхностей  $\vec{P}_{rabbit}(u, v)$  и  $\vec{P}_{fox}(u, v)$  соответственно построим динамические локальные базисы  $[\vec{E}_1 \quad \vec{E}_2 \quad \vec{E}_3]$  и  $[\vec{Q}_1 \quad \vec{Q}_2 \quad \vec{Q}_3]$  (рис. 1), где  $\vec{E}_1 = \frac{1}{V_{rabbit}} \times \frac{d\vec{R}_{rabbit}(T)}{dT}$  и  $\vec{Q}_1 = \frac{1}{V_{fox}} \times \frac{d\vec{R}_{fox}(T)}{dT}$ .  $V_{rabbit}$  и  $V_{fox}$  – постоянные по модулю скорости «Кролика» и «Лисы».  $\vec{E}_3$  и  $\vec{Q}_3$  – единичные нормали  $\vec{n}_{rabbit}$ ,  $\vec{n}_{fox}$  в точках  $\vec{R}_{rabbit}(T)$  и  $\vec{R}_{fox}(T)$ , и  $\vec{E}_2 = \vec{E}_3 \times \vec{E}_1$ ,  $\vec{Q}_2 = \vec{Q}_3 \times \vec{Q}_1$ .

Далее будем считать, что плоскость  $\Sigma$  образована векторами  $\vec{E}_1$  и  $\vec{E}_2$ , а плоскость  $\Pi$  – векторами  $\vec{Q}_1$  и  $\vec{Q}_2$ . Для выбора решения «Кроликом» нам надо спроектировать точку  $\vec{R}_{fox}$  на плоскость  $\Sigma$  и получить ее проекцию  $\vec{R}'_{fox}$  для анализа, на предмет из какой четверти плоскости  $\Sigma$  приближается проекция «Лисы». Точно также для принятия решения «Лисой» необхо-

димо спроектировать точку  $\vec{R}_{rabbit}$  на плоскость  $\Pi$ , получить ее проекцию  $\vec{R}'_{rabbit}$ , для того чтобы проанализировать, в какой четверти плоскости  $\Pi$  она находится (рис. 1).

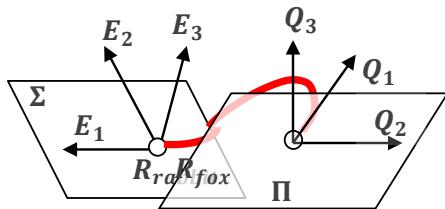


Рис. 1. Динамические базисы

Итак, целью статьи является детальная разработка алгоритма и написание программы в системе компьютерной математики MathCAD для указанной выше задачи преследования.

### Моделирование поверхностей

Моделирование поверхности  $\vec{P}_{rabbit}(u, v)$ , по которой движется «Кролик», производится заданием точек линий горизонталей в среде AutoCAD с последующей полиномиальной регрессией в среде MathCAD при помощи встроенных функций математического пакета. На полученной после полиномиальной регрессии поверхности вводится равномерная сетка с расчетом частных производных в узлах сетки.

С моделью поверхности, по которой движется «Лиса», в нашей программе дело обстоит так. К поверхности «Кролика»  $\vec{P}_{rabbit}(u, v)$  строится эквидистантная поверхность (рис. 2)  $\vec{P}_{equ}(u, v) = \vec{P}_{rabbit}(u, v) + R_{equ} \times \vec{n}_{rabbit}$ , где  $R_{equ}$  – расстояние, на которое эквидистантная поверхность отстоит от исходной.

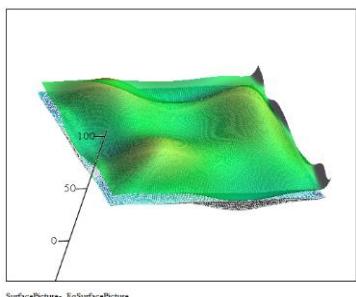


Рис. 2. Эквидистантная поверхность

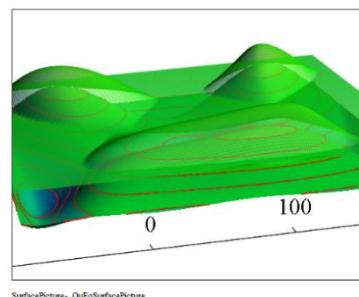


Рис. 3. «Квазиеквидистантная» поверхность

Потом вводится горизонтальная плоскость  $z = r_0$ , и поверхность  $\vec{P}_{fox}(u, v)$  составляется следующим образом:  $\vec{P}_{fox}(u, v) = \begin{cases} z, & \text{если } \vec{P}_{equ}(u, v)_z \leq r_0 \\ \vec{P}_{equ}(u, v), & \text{если } \vec{P}_{equ}(u, v)_z > r_0 \end{cases}$ , где  $r_0$  – некоторое пороговое значение высоты (рис. 3). После этого на поверхности  $\vec{P}_{fox}(u, v)$  вводится равномерная сетка с расчетом частных производных в узлах. На рис. 3 поверхность  $\vec{P}_{fox}(u, v)$  представлена в полупрозрачном виде, а исходная поверхность «Кролика» представлена с обозначением горизонталей.

### Построение проекций в локальных базисах

Если «Кролик», двигаясь по поверхности  $\vec{P}_{rabbit}(u, v)$ , образует динамический базис  $[\vec{E}_1 \quad \vec{E}_2 \quad \vec{E}_3]$ , то плоскость  $\Sigma$ , образованная векторами  $\vec{E}_1$  и  $\vec{E}_2$ , является касательной пло-

стью к поверхности  $\vec{P}_{rabbit}(u, v)$  в точке  $\vec{R}_{rabbit}$ . Проекция точки  $\vec{R}_{fox}$  на плоскость  $\Sigma$  будет такой:  $\vec{R}'_{fox} = \vec{R}_{fox} + \frac{\vec{E}_3 \times (\vec{R}_{rabbit} - \vec{R}_{fox})}{\vec{E}_3 \times \vec{E}_3}$  (рис. 4). «Лиса», двигаясь по поверхности  $\vec{P}_{fox}(u, v)$ , образует динамический базис  $[\vec{Q}_1 \quad \vec{Q}_2 \quad \vec{Q}_3]$ . Проекция точки  $\vec{R}_{rabbit}$  на плоскость  $\Pi$ , образованную векторами  $\vec{Q}_1$  и  $\vec{Q}_2$ , будет такой:  $\vec{R}'_{rabbit} = \vec{R}_{rabbit} + \frac{\vec{Q}_3 \times (\vec{R}_{fox} - \vec{R}_{rabbit})}{\vec{Q}_3 \times \vec{Q}_3}$  (рис. 5).

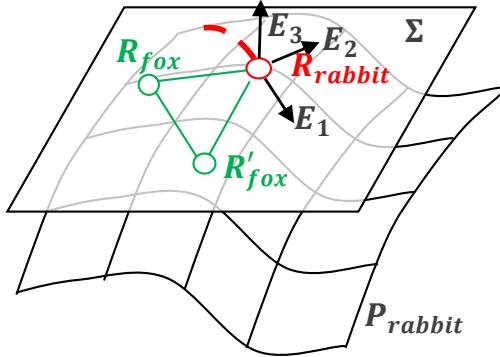


Рис. 4. Проекция «Лисы»

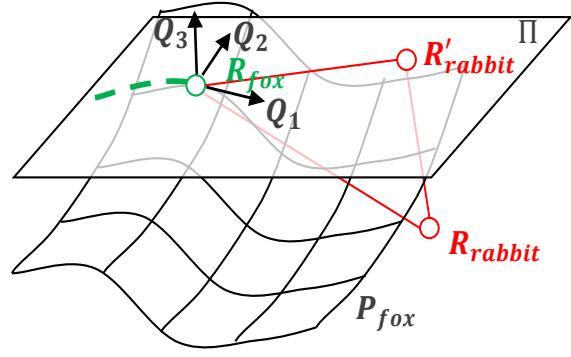


Рис. 5. Проекция «Кролика»

Тогда в базисе  $[\vec{E}_1 \quad \vec{E}_2 \quad \vec{E}_3]$  локальной системы координат «Кролика» координаты

проекции «Лисы»  $\vec{R}'_{fox}$  будут такими:  $\vec{R}'_{fox_\Sigma} = \begin{bmatrix} (\vec{R}'_{fox} - \vec{R}_{rabbit}) \times \vec{E}_1 \\ (\vec{R}'_{fox} - \vec{R}_{rabbit}) \times \vec{E}_2 \\ (\vec{R}'_{fox} - \vec{R}_{rabbit}) \times \vec{E}_3 \end{bmatrix}$ . В локальной системе

координат  $[\vec{Q}_1 \quad \vec{Q}_2 \quad \vec{Q}_3]$  «Лисы» координаты проекции «Кролика»  $\vec{R}'_{rabbit}$  будут такими:

$\vec{R}'_{rabbit_\Pi} = \begin{bmatrix} (\vec{R}'_{rabbit} - \vec{R}_{fox}) \times \vec{Q}_1 \\ (\vec{R}'_{rabbit} - \vec{R}_{fox}) \times \vec{Q}_2 \\ (\vec{R}'_{rabbit} - \vec{R}_{fox}) \times \vec{Q}_3 \end{bmatrix}$ . Необходимо отметить, что  $(\vec{R}'_{fox} - \vec{R}_{rabbit}) \times \vec{E}_3 = 0$ ,

$$(\vec{R}'_{rabbit} - \vec{R}_{fox}) \times \vec{Q}_3 = 0.$$

Базис мировой системы координат  $[\vec{H}_1 \quad \vec{H}_2 \quad \vec{H}_3]$  системе координат «Кролика»

будет выглядеть так:  $\vec{H}_\Sigma = \begin{bmatrix} \vec{H}_{1\Sigma} \\ \vec{H}_{2\Sigma} \\ \vec{H}_{3\Sigma} \end{bmatrix}$ , где  $\vec{H}_{1\Sigma} = \begin{bmatrix} \vec{H}_1 \times \vec{E}_1 \\ \vec{H}_1 \times \vec{E}_2 \\ \vec{H}_1 \times \vec{E}_3 \end{bmatrix}$ ,  $\vec{H}_{2\Sigma} = \begin{bmatrix} \vec{H}_2 \times \vec{E}_1 \\ \vec{H}_2 \times \vec{E}_2 \\ \vec{H}_2 \times \vec{E}_3 \end{bmatrix}$ ,  $\vec{H}_{3\Sigma} = \begin{bmatrix} \vec{H}_3 \times \vec{E}_1 \\ \vec{H}_3 \times \vec{E}_2 \\ \vec{H}_3 \times \vec{E}_3 \end{bmatrix}$ ,

а в системе координат «Лисы» так:  $\vec{H}_\Pi = \begin{bmatrix} \vec{H}_{1\Pi} \\ \vec{H}_{2\Pi} \\ \vec{H}_{3\Pi} \end{bmatrix}$ , где  $\vec{H}_{1\Pi} = \begin{bmatrix} \vec{H}_1 \times \vec{E}_1 \\ \vec{H}_1 \times \vec{E}_2 \\ \vec{H}_1 \times \vec{E}_3 \end{bmatrix}$ ,  $\vec{H}_{2\Pi} = \begin{bmatrix} \vec{H}_2 \times \vec{E}_1 \\ \vec{H}_2 \times \vec{E}_2 \\ \vec{H}_2 \times \vec{E}_3 \end{bmatrix}$ ,

$$\vec{H}_{3\Pi} = \begin{bmatrix} \vec{H}_3 \times \vec{E}_1 \\ \vec{H}_3 \times \vec{E}_2 \\ \vec{H}_3 \times \vec{E}_3 \end{bmatrix}.$$

### Выбор направления движения

Нами написана программа, которая в расчетном цикле на каждом шаге делает выбор, в каком направлении двигаться «Кролику» и «Лисе». Пусть  $\omega_{\text{rabit}}$  и  $\omega_{\text{fox}}$  – угловые скорости «Кролика» и «Лисы». Для примера рассмотрим выбор направления движения «Кроликом».

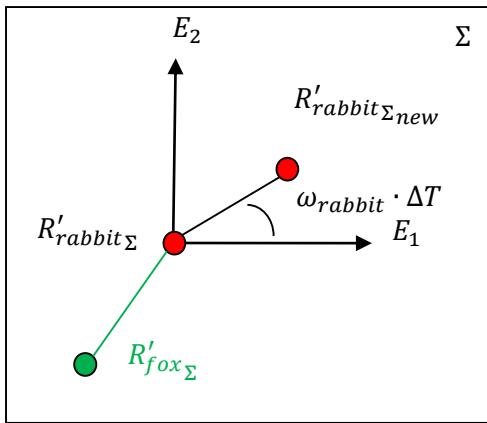


Рис. 6. Выбор направления движения «Кроликом»

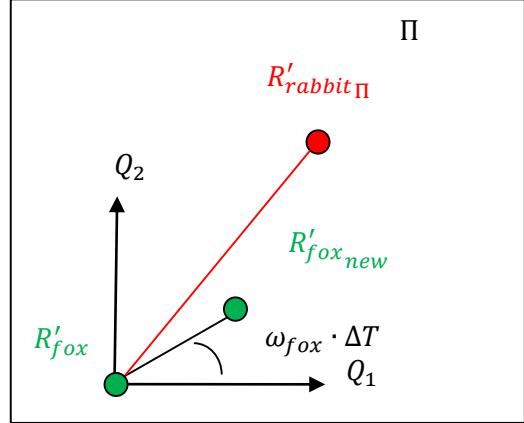


Рис. 7. Выбор направления движения «Лисой»

На плоскости  $\Sigma$  в системе координат «Кролика» координаты «Кролика» равны  $\vec{R}'_{rabbit\Sigma} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ . Анализ координат «Лисы» «Кроликом» состоит в том, что если «Лиса» находится в нижней полуплоскости  $\Sigma$ , то «Кролик» совершает шаг против часовой стрелки (рис. 6). Если «Лиса» находится в верхней полуплоскости, то «Кролик» совершает шаг по часовой, то есть новые координаты «Кролика» за промежуток времени  $\Delta T$  равны в системе координат плоскости  $\Sigma$ :

$$\vec{R}'_{rabbit\Sigma new} = \begin{cases} V_{rabbit} \times \Delta T \times \begin{bmatrix} \cos(\omega_{rabbit} \times \Delta T) \\ \sin(\omega_{rabbit} \times \Delta T) \\ 0 \end{bmatrix}, & \text{if } (\vec{R}'_{fox} - \vec{R}_{rabbit}) \times \vec{E}_2 < 0 \\ V_{rabbit} \times \Delta T \times \begin{bmatrix} \cos(\omega_{rabbit} \times \Delta T) \\ -\sin(\omega_{rabbit} \times \Delta T) \\ 0 \end{bmatrix}, & \text{if } (\vec{R}'_{fox} - \vec{R}_{rabbit}) \times \vec{E}_2 \geq 0 \end{cases}.$$

Так же можно сказать про выбор направления движения «Лисой». Если «Кролик» находится в верхней полуплоскости  $\Pi$ , то «Лиса» совершает шаг против часовой (рис. 7). Если в нижней полуплоскости, то по часовой стрелке:

$$\vec{R}'_{fox\Pi new} = \begin{cases} V_{fox} \times \Delta T \times \begin{bmatrix} \cos(\omega_{fox} \times \Delta T) \\ -\sin(\omega_{fox} \times \Delta T) \\ 0 \end{bmatrix}, & \text{if } (\vec{R}'_{rabbit} - \vec{R}_{fox}) \times \vec{Q}_2 < 0 \\ V_{fox} \times \Delta T \times \begin{bmatrix} \cos(\omega_{fox} \times \Delta T) \\ \sin(\omega_{fox} \times \Delta T) \\ 0 \end{bmatrix}, & \text{if } (\vec{R}'_{rabbit} - \vec{R}_{fox}) \times \vec{Q}_2 \geq 0 \end{cases}.$$

#### Расчет координат на следующих этапах итераций

Ранее мы рассматривали, как будет выглядеть базис  $[\vec{H}_1 \quad \vec{H}_2 \quad \vec{H}_3]$  мировой системы координат в локальных динамических базисах  $[\vec{E}_1 \quad \vec{E}_2 \quad \vec{E}_3]$  и  $[\vec{Q}_1 \quad \vec{Q}_2 \quad \vec{Q}_3]$ . Тогда в мировой системе координат  $[\vec{H}_1 \quad \vec{H}_2 \quad \vec{H}_3]$  новые координаты «Кролика»  $\vec{R}'_{rabbit\Sigma new}$  на плоскости  $\Sigma$  будут такими:

$$\vec{R}'_{rabbit new} = \vec{R}_{rabbit} + \begin{bmatrix} \vec{R}'_{rabbit\Sigma new} \times \vec{H}_{1\Sigma} \\ \vec{R}'_{rabbit\Sigma new} \times \vec{H}_{2\Sigma} \\ \vec{R}'_{rabbit\Sigma new} \times \vec{H}_{3\Sigma} \end{bmatrix}.$$

А новые координаты «Лисы»  $\vec{R}'_{fox_{\Pi_{new}}}$  на плоскости  $\Pi$  на том же этапе итераций будут выглядеть в мировой системе координат так:

$$\vec{R}'_{fox_{new}} = \vec{R}_{fox} + \begin{bmatrix} \vec{R}'_{fox_{\Pi_{new}}} \times \vec{H}_{1\Pi} \\ \vec{R}'_{fox_{\Pi_{new}}} \times \vec{H}_{2\Pi} \\ \vec{R}'_{fox_{\Pi_{new}}} \times \vec{H}_{3\Pi} \end{bmatrix}.$$

Нам нужны вертикальные проекции точек  $\vec{R}'_{rabbit_{new}}$ ,  $\vec{R}'_{fox_{new}}$  на поверхности  $\vec{P}_{rabbit}(u, v)$  и  $\vec{P}_{fox}(u, v)$  соответственно. Поверхности  $\vec{P}_{rabbit}(u, v)$  и  $\vec{P}_{fox}(u, v)$ , по которым движутся «Кролик» и «Лиса», всегда можно преобразовать к виду:  $\vec{P}_{rabbit}(x, y) = \begin{bmatrix} x \\ y \\ S_{rabbit}(x, y) \end{bmatrix}$  и  $\vec{P}_{fox}(x, y) = \begin{bmatrix} x \\ y \\ S_{fox}(x, y) \end{bmatrix}$ . Если координаты точек  $\vec{R}'_{rabbit_{new}}$  и  $\vec{R}'_{fox_{new}}$ , то мы

представим их в ином виде:  $\vec{R}'_{rabbit_{new}} = \begin{bmatrix} x_{rabbit_{new}} \\ y_{rabbit_{new}} \\ z_{rabbit_{new}} \end{bmatrix}$ ,  $\vec{R}'_{fox_{new}} = \begin{bmatrix} x_{fox_{new}} \\ y_{fox_{new}} \\ z_{fox_{new}} \end{bmatrix}$ . Тогда искомые вертикальные проекции точек  $\vec{R}'_{rabbit_{new}}$ ,  $\vec{R}'_{fox_{new}}$  на поверхности можно представить в виде:

$$\vec{R}_{rabbit_{new}} = \begin{bmatrix} x_{rabbit_{new}} \\ y_{rabbit_{new}} \\ S_{rabbit}(x_{rabbit_{new}}, y_{rabbit_{new}}) \end{bmatrix}, \vec{R}_{fox_{new}} = \begin{bmatrix} x_{fox_{new}} \\ y_{fox_{new}} \\ S_{fox}(x_{fox_{new}}, y_{fox_{new}}) \end{bmatrix}.$$

Таким образом, мы считаем итерационный процесс в поставленной нами задаче преследования сформированным. Критерием достижения цели в разработанной нами программе служит расстояние между «Кроликом» и «Лисой» в горизонтальной плоскости проекций:

$$\sqrt{(x_{rabbit_{new}} - x_{fox_{new}})^2 + (y_{rabbit_{new}} - y_{fox_{new}})^2} = R_0, \text{ где } R_0 \text{ – некоторое заданное нами пороговое значение.}$$

## Результаты и выводы

В данной статье рассматривалась задача преследования, в которой участники находятся на разных поверхностях. Теоретические аспекты задачи о «водителе-убийце» («катер-торпеда») были рассмотрены в работах Руфуса Айзекса [1]. В работах Л. С. Понtryгина рассматривались теоретические вопросы задачи преследования в пространстве (самолеты) [2]. В работе [3] рассматривалась линеаризованная задача преследования на плоскости с постоянными скоростями и ограниченной кривизной траекторий. В работе [4] рассматривались алгоритмические вопросы управления мобильными роботами в задаче преследования на плоскости.

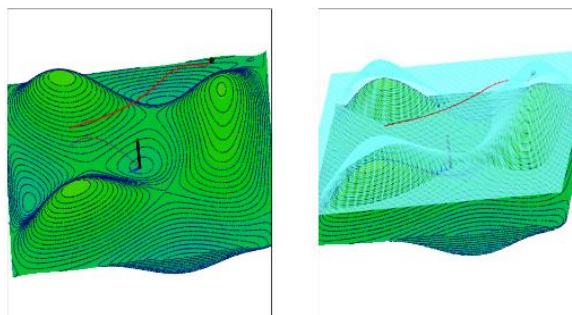


Рис. 8. Преследование по «квазиэквидистанте»

В монографии [5] тоже были предложены алгоритмы для осуществления позиционных стратегий. В работе [6] рассматривались модели движения с ограниченной кривизной траекторий движения.

На основе работ [1–6] предложена математическая модель задачи преследования, когда участники задачи находятся на разных поверхностях. Приближенно можно считать аналогом преследования с воздуха по траектории, учитывающей рельеф местности. Разработан алгоритм построения траекторий в зависимости от пространственного расположения оппонентов. Предложен метод анализа координат оппонента и принятия решений о выборе направления движения.

В системе компьютерной математики MathCAD разработана программа, реализующая данную модель задачи преследования [7]. По результатам работы программы изготовлено анимированное изображение [8].

На рис. 8 представлен первый кадр процесса преследования. В левой части рисунка поверхность «Лисы» не отображена, видимыми являются только траектории «Лисы», «Кролика» и поверхность «Кролика». В правой части рисунка поверхность «Лисы» отображена в полупрозрачном виде.

По мнению автора, результаты исследования, изложенные в данной статье, могут быть востребованы разработчиками робототехнических комплексов с элементами искусственного интеллекта.

Работа выполнена при поддержке проекта конкурса инновационных грантов Бурятского государственного университета 2019 г. на создание аппаратно-программного комплекса сопровождения.

### **Литература**

1. Айзекс Р. Дифференциальные игры. М. : Мир, 1967.
2. Понtryгин Л. С. Линейная дифференциальная игра убегания. Тр. МИАН СССР. 1971. Т. 112. С. 30–63.
3. Желнин Ю. Н. Линеаризованная задача преследования и уклонения на плоскости // Ученые записки ЦАГИ. № 3. Т. 8. 1977. С. 88–98.
4. Бурдаков С. В., Сизов П. А. Алгоритмы управлением движения мобильным роботом в задаче преследования // Научн.-техн. ведомости Санкт-Петербург. гос. политехн. ун-та. Информатика. Телекоммуникации. Управление. 2014. № 6 (210). С. 49–58.
5. Красовский Н. Н., Субботин А. И. Позиционные дифференциальные игры. М. : Наука, 1974 г. 456 с.
6. Симакова Э. Н. Об одной дифференциальной игре преследования // Автоматика и телемеханика. 1967. № 2. С. 5–14.
7. Движение «Лисы» по «квазиэвидистанте». URL: <http://dubanov.exponenta.ru> (дата обращения: 28.02.2019).
8. Движение «Лисы» по «квазиэвидистанте». URL: <https://www.youtube.com> (дата обращения: 28.02.2019).