

**Н.Л. Григоренко, Д.В. Камзолкин
Д.Г. Пивоварчук**

**ЛИНЕЙНЫЕ
ДИФФЕРЕНЦИАЛЬНЫЕ ИГРЫ
Численные методы решения**

**Методическая разработка для студентов
4 курса**

2007

Содержание

1	Цели и задачи практикума	3
2	Постановка задачи	5
3	Теоретические основы решения	10
4	Метод решения	13
4.1	Вычисление альтернированной суммы	15
4.2	Гарантированное время окончания преследования	16
4.3	Позиционные стратегии преследования и уклонения	16
5	Вычисление геометрической разности двух выпуклых компактов	19
5.1	Вычисление выпуклой оболочки положительно однородной функции	21
5.2	Алгоритм приближенного вычисления геометрической разности	24
5.3	Визуализация геометрической разности и проверка на пустоту	25
6	Алгоритмы построения многомерных сеток	25
6.1	Построение сетки на поверхности единичной сферы для 2-х мерного случая	26
6.2	Построение сетки на поверхности единичной гипersферы в общем случае	26
7	Примеры для тестирования программы	27
7.1	Два Крокодила R^2	28
7.2	Контрольный пример Понтрягина R^2	28
7.3	Два Крокодила R^3	29
7.4	Контрольный пример Понтрягина R^3	30
8	Опорные функции некоторых множеств в R^n	31
9	Дополнительные задания	31
10	Символьные вычисления	43
	Список литературы	45

1 Цели и задачи практикума

Математический практикум "Линейные дифференциальные игры: численные методы решения" заключается в разработке компьютерной программы для решения линейных стационарных дифференциальных игр методом альтернированного интеграла Л.С. Понтрягина [1, 2, 3]. Практикум состоит из двух частей: теоретической и практической.

Первая часть заключается в изучении теоретических основ задания: постановки линейной дифференциальной игры наведения-уклонения, определения и свойств альтернированного интеграла и алгоритмов и методов его построения при помощи опорных функций выпуклых множеств, достаточных условий разрешимости задач наведения и уклонения, способов построения позиционных стратегий игроков.

Практическая часть состоит из составления программы на алгоритмическом языке, решающей в общем виде линейную дифференциальную игру, отладки программы, решении тестовых примеров и индивидуального задания.

Завершается работа предоставлением отчета.

Цель задания

- Знакомство с численными методами решения линейных дифференциальных игр.
- Практика использования численных методов.
- Разработка компьютерной программы решения линейных дифференциальных игр с помощью альтернированного интеграла Л.С. Понтрягина.
- Расчет примеров.
- Накопление опыта по созданию программных средств для решения задач управления.

Содержание задания

- Изучение теоретических основ построения альтернированного интеграла для решения линейных дифференциальных игр.
- Составление и отладка программы для численного решения линейных дифференциальных игр указанным выше методом.

- Решение конкретных примеров задач.
- Написание отчета о проделанной работе.

Требования к программе

Разрабатываемая программа должна быть снабжена дружественным пользователю интерфейсом для ввода постановки линейной дифференциальной игры в общем виде и параметров численного метода, контроля за состоянием расчетов, вывода результатов решения.

Исходный текст программы, отвечающий за реализацию численного метода, должен быть снабжен комментариями. Ввод новой постановки дифференциальной игры и изменение параметров численных методов, а также вывод требуемых результатов не должен требовать изменения исходного кода. В программе должна быть реализована возможность сохранения введенной дифференциальной игры в файл. Ввод функций в программу осуществляется в символьном виде, множества задаются с помощью их опорных функций и предполагаются выпуклыми. Обязательно наличие возможности останова вычислений для выхода из программы или корректировки входных данных.

Вывод результатов осуществляется как в графическом, так и в табличном виде (визуализация альтернированного интеграла осуществляется только в 2-х мерном случае). Должна быть предусмотрена возможность вывода графиков фазовых переменных и управлений игроков как в одно, так и в отдельные окна. Все графики и рисунки должны иметь подписи на осях.

Содержание отчета

- Титульный лист с идентификацией студента и задания.
- Постановка задачи и краткое описание метода решения.
- Часть исходного кода программы, отвечающая за реализацию численного метода построения альтернированного интеграла Л.С. Понтрягина, с комментариями.
- Описание интерфейса программы и последовательности действий, требуемых для решения линейной дифференциальной игры, начиная от ввода исходных данных и кончая выводом результатов вычислений.
- Результаты решения тестовых примеров, имеющих аналитическое решение, и сравнение полученного решения с точным.

- Результат решения индивидуального задания.

Этапы выполнения задания

Задание практикума рассчитано на один учебный семестр (14 недель) и состоит из следующих этапов.

	Наименование этапа	Срок выполнения
1	Разработка интерфейса для ввода параметров дифференциальной игры и вывода результатов.	1 неделя
2	Вычисление альтернированных сумм. Вывод результата в табличной форме	4 недели
3	Визуализация альтернированных сумм для случая размерности 2.	2 недели
4	Вычисление гарантированного времени решения задачи наведения.	1 неделя
5	Решение тестовых примеров. Отладка программы.	2 недели
6	Построение стратегии первого игрока, решающей задачу наведения.	1 неделя
7	Построение стратегии второго игрока, решающей задачу уклонения.	1 неделя
8	Решение индивидуального задания. Предоставление итогового отчета.	2 недели

2 Постановка задачи

Задана линейная стационарная управляемая динамическая система, описываемая на отрезке времени $t \in [0, T]$ дифференциальным уравнением

$$\dot{x}(t) = Ax(t) - Bu(t) + Cv(t), \quad (1)$$

где $x(t) \in R^n$ – фазовый вектор, описывающий состояние системы, $u(t) \in P \subset R^p$, $v(t) \in Q \subset R^q$ – управления первого и второго игроков соответственно. Множества P и Q – выпуклые компакты, называемые ресурсами управления первого и второго игроков. $A \subset R^{n \times n}$, $B \subset R^{n \times p}$ и $C \subset R^{n \times q}$ – постоянные матрицы.

Определение 1. Пара $(t, x) \in [0, T] \times R^n$ называется *позицией управляемой системы (1)*.

Определение 2. Допустимым программным управлением первого игрока называется любая кусочно-непрерывная функция $u(\cdot)$, определенная на отрезке $[0, T]$ и удовлетворяющая на нем геометрическому ограничению $u(t) \in P$.

Допустимым программным управлением второго игрока называется любая кусочно-непрерывная функция $v(\cdot)$, определенная на отрезке $[0, T]$ и удовлетворяющая на нем геометрическому ограничению $v(t) \in Q$.

Обозначим класс всех допустимых программных управлений первого игрока, определенных на отрезке $[0, T]$, через $\mathcal{U}[0, T]$, а класс всех допустимых программных управлений второго игрока, определенных на отрезке $[0, T]$, через $\mathcal{V}[0, T]$.

Пусть задано начальное условие

$$x(0) = x_0,$$

выпуклое компактное терминальное множество $M \subset R^\nu$ и матрица $\pi \subset R^{\nu \times n}$, где $\nu \leq n$.

Пусть далее выбраны допустимые управления $u(\cdot)$ и $v(\cdot)$ первого и второго игроков соответственно.

Определение 3. Движением управляемой системы (1), исходящим из начальной позиции $(0, x_0)$ под действием допустимых управлений $u(\cdot)$ первого игрока и $v(\cdot)$ второго игрока, называется решение дифференциального уравнения (1) с начальным условием $x(0) = x_0$, определенное на отрезке $[0, T]$.

В силу линейности управляемой системы такое движение всегда существует и единственно.

Целью первого игрока (задача наведения на терминальное множество) является приведение проекции траектории $x(t)$ управляемой системы (1) в конечный момент времени T на терминальное множество M , то есть выполнение включения

$$\pi x(T) \in M. \quad (2)$$

Целью второго игрока (задача уклонения от терминального множества) противоположна

$$\pi x(T) \notin M. \quad (3)$$

Соответственно рассматриваются две задачи: задача наведения и задача уклонения, составляющие вместе дифференциальную игру. Для

формального описания этих задач необходимо определить классы допустимых управлений игроков, в рамках которых они решают свои задачи.

Определение 4. *Разбиением отрезка времени $[0, T]$ называется всякое конечное семейство вида:*

$$\omega_N = \{t_i\}_{i=0}^N : 0 = t_0 < t_1 < \dots < t_N = T,$$

при этом величина $d(\omega_N) = \max_{i=0, \dots, N-1} (t_{i+1} - t_i)$ – называется *диаметром разбиения*.

Определим сначала задачу наведения для первого игрока. Пусть он по определенному закону может изменять свое управление только в определенные моменты времени из некоторого разбиения временного отрезка $[0, T]$, оставляя управление постоянным между этими моментами.

Определение 5. *Позиционной стратегией первого игрока называется любая функция $U(\cdot): (t, x) \rightarrow U(t, x): [0, T] \times R^n \rightarrow P$.*

Принадлежность значений функции $U(\cdot)$ множеству P является единственным ограничением.

Обозначим класс всех позиционных стратегий первого игрока через $\mathcal{U}_{pos}[0, T]$.

Движение системы (1), под действием позиционного управления $U \in \mathcal{U}_{pos}[0, T]$ первого игрока, определяется следующим образом. Пусть первый игрок изменяет свое управление только в моменты t_i разбиения ω_N , оставляя его постоянным между этими моментами.

Определение 6. *Движением управляемой системы (1), исходящим из начальной позиции $(0, x_0)$ под действием позиционной стратегии U первого игрока и управления $v(\cdot)$ второго игрока на разбиении ω_N , называется решение $x(\cdot)$ дифференциального уравнения (1) с начальным условием $x(0) = x_0$, где $u(t) = U(t_i, x(t_i))$ ($t \in [t_i, t_{i+1})$, $i = 0, \dots, N-1$).*

Данное движение существует, единственно и обозначается $x_{\omega_N}(\cdot | 0, x_0, U, v(\cdot))$.

Определение 7. *Пучком движений из заданной позиции $(0, x_0)$, порожденным позиционной стратегией $U \in \mathcal{U}_{pos}[0, T]$ первого игрока на разбиении ω_N , называется множество*

$$X_{\omega_N}(0, x_0, U) = \{x_{\omega_N}(\cdot | 0, x_0, U, v(\cdot)) : v(\cdot) \in \mathcal{V}[0, T]\}.$$

Задача наведения из позиции $(0, x_0)$ на терминальное множество M в момент времени T в классе позиционных стратегий первого игрока заключается в отыскании такой позиционной стратегии $U \in \mathcal{U}_{pos}[0, T]$ первого игрока, что для любого $\varepsilon > 0$ найдется такое $\delta > 0$, что для любого разбиения ω_N отрезка $[0, T]$ с диаметром $d(\omega_N) < \delta$ и для каждого движения $x(\cdot) \in X_{\omega_N}(0, x_0, U)$ справедливо $\pi x(T) \in M^\varepsilon$. Здесь M^ε – ε -окрестность множества M .

Аналогичным образом определяется задача для второго игрока.

Определение 8. *Позиционной стратегией второго игрока называется любая функция $V(\cdot): (t, x) \rightarrow V(t, x): [0, T] \times R^n \rightarrow Q$.*

Обозначим класс всех позиционных стратегий второго игрока через $\mathcal{V}_{pos}[0, T]$.

Определение 9. *Движением управляемой системы (1), исходящим из начальной позиции $(0, x_0)$ под действием позиционной стратегии V второго игрока и управления $u(\cdot)$ первого игрока на разбиении ω_N , называется решение $x(\cdot)$ дифференциального уравнения (1) с начальным условием $x(0) = x_0$, где $v(t) = V(t_i, x(t_i))$ ($t \in [t_i, t_{i+1})$, $i = 0, \dots, N-1$).*

Данное движение обозначается $x_{\omega_N}(\cdot|0, x_0, u(\cdot), V)$.

Определение 10. *Пучком движений из заданной позиции $(0, x_0)$, порожденным позиционной стратегией $V \in \mathcal{V}_{pos}[0, T]$ второго игрока на разбиении ω_N , называется множество*

$$X_{\omega_N}(0, x_0, V) = \{x_{\omega_N}(\cdot|0, x_0, u(\cdot), V) : u(\cdot) \in \mathcal{U}[0, T]\}.$$

Задача уклонения из позиции $(0, x_0)$ от терминального множества M в момент времени T в классе позиционных стратегий второго игрока заключается в отыскании такой позиционной стратегии $V \in \mathcal{V}_{pos}[0, T]$ второго игрока, что существуют $\varepsilon > 0$ и $\delta > 0$ такие, что для любого разбиения ω_N отрезка $[0, T]$ с диаметром $d(\omega_N) < \delta$ и для каждого движения $x(\cdot) \in X_{\omega_N}(0, x_0, V)$ справедливо $\pi x(T) \notin M^\varepsilon$.

Справедлива следующая теорема.

Теорема 1. *Задачи наведения и уклонения в классах позиционных стратегий игроков из одной начальной позиции одновременно не разрешимы.*

Ниже приведен пример использования позиционных стратегий и построения соответствующих движений.

Пример.

Рассмотрим одномерную управляемую систему

$$\dot{x}(t) = u(t) + v(t),$$

с начальным условием $x(0) = 0$ и конечным моментом времени $T = 1$. Управления выбираются из множеств $P = [-1, 1]$ и $Q = [-1, 1]$, и пусть $\pi = 1$. Терминальное множество представляет собой точку $M = \{0\}$. Покажем, что позиционная стратегия

$$U(t, x) = \begin{cases} -1, & \text{при } x > 0, \\ 1, & \text{при } x \leq 0 \end{cases}$$

решает задачу наведения из позиции $(0, 0)$.

Рассмотрим произвольное разбиение $\omega_N = \{t_i\}_{i=0}^N$ отрезка $[0, 1]$ с диаметром δ и произвольное управление $v(\cdot)$ второго игрока. Покажем, что если в момент t_i ($i = 0, \dots, N - 1$) выполняется условие $x(t_i) \in (-2\delta, 2\delta]$, то и $x(t_{i+1}) \in (-2\delta, 2\delta]$. Если $x(t_i) \in (0, 2\delta]$, то на полуинтервале $[t_i, t_{i+1})$ первый игрок использует постоянное управление $u_i = U(t_i, x(t_i)) = -1$. Следовательно

$$x(t_{i+1}) = x(t_i) + \int_{t_i}^{t_{i+1}} (-1 + v(t)) dt$$

и из ограниченности управления второго игрока $-2\delta < x(t_{i+1}) \leq 2\delta$.

Если же $x(t_i) \in (-2\delta, 0]$, то на полуинтервале $[t_i, t_{i+1})$ первый игрок использует постоянное управление $u_i = 1$. Следовательно

$$x(t_{i+1}) = x(t_i) + \int_{t_i}^{t_{i+1}} (1 + v(t)) dt$$

и значит $-2\delta < x(t_{i+1}) \leq 2\delta$.

Поскольку $x(t_0) = 0$, то получаем для любого $i = 0, \dots, N$ оценку $-2\delta < x(t_i) \leq 2\delta$ и следовательно

$$-2\delta < x(T) \leq 2\delta.$$

Выбирая теперь $\delta < \frac{\varepsilon}{2}$, получаем для движения $x(\cdot)$ включение $x(T) \in M^\varepsilon$. То есть позиционная стратегия U решает задачу наведения.

Пусть зафиксирован временной отрезок $[0, \Theta]$. Задание практикума состоит из двух частей. Первая часть заключается в проверке достаточных условий существования гарантированного времени $T \in [0, \Theta]$ решения задачи наведения, построении стратегии $U^*(t, x)$ первого игрока, решающей эту задачу, и стратегии $V^*(t, x)$ второго игрока, обеспечивающей максимальное противодействие. Под стратегией максимального противодействия понимается стратегия, исключающая попадание проекции траектории системы на терминальное множество M до гарантированного момента T .

Вторая часть заключается в проверке достаточных условий решения задачи уклонения в заданный момент времени Θ и построении стратегии $V^*(t, x)$ второго игрока, обеспечивающей это уклонение.

Для решения дифференциальной игры предлагается использовать метод альтернированного интеграла Л.С. Понтрягина [1].

3 Теоретические основы решения

Выпишем достаточные условия разрешимости задач наведения и уклонения для заданного конечного момента времени $T \in [0, \Theta]$. Рассматривается замена фазовой переменной

$$z(t) = \pi e^{(T-t)A} x(t) \in R^\nu.$$

Управляемая система (1) сводится к системе с простой динамикой

$$\dot{z}(t) = -\pi e^{(T-t)A} B u(t) + \pi e^{(T-t)A} C v(t). \quad (4)$$

Начальное условие примет вид

$$z(0) = \pi e^{TA} x_0.$$

Условие (2) преобразуется во включение

$$z(T) \in M,$$

а условие (3) во включение

$$z(T) \notin M.$$

Введем ряд определений, которые будут нам полезны в дальнейшем. Дополнительная информация по выпуклому анализу содержится в работе [11].

Определение 11. Пусть $X \subset R^n$ – произвольное множество, а $\lambda > 0$ – произвольное действительное число. Произведением множества на число называется множество

$$\lambda X = \{\lambda x : x \in X\}.$$

Определение 12. Пусть $X_1, X_2 \subset R^n$ – произвольные множества. Их алгебраической суммой называется множество

$$X_1 + X_2 = \{x_1 + x_2 : x_1 \in X_1, x_2 \in X_2\},$$

а геометрической разностью множество

$$X_1 * X_2 = \{x \in R^n : x + X_2 \subset X_1\}.$$

Если множества X_1 и X_2 – выпуклы и компактны, то их сумма и разность тоже будут выпуклыми компактными множествами.

Определение 13. Пусть на отрезке $[t_0, t_1]$ задано непрерывное многозначное отображение $X(t)$, значениями которого являются непустые компактные множества из пространства R^n . Интегралом от многозначного отображения $X(t)$ называется множество

$$\int_{t_0}^{t_1} X(t)dt = \left\{ \int_{t_0}^{t_1} x(t)dt : x(t) \in X(t) \right\}.$$

Здесь в правой части интеграл берется по все функциям $x(t)$, для которых он существует.

Интеграл от многозначного отображения является непустым выпуклым компактным множеством.

Перейдем теперь непосредственно к определению альтернированного интеграла. Для решения дифференциальной игры выбирается равномерное разбиение $\omega_N = \{t_i\}_{i=0}^N$ отрезка $[0, T]$ с диаметром $d(\omega_N) = T/N$ и строится альтернированная сумма $W_N^N \subset R^\nu$, определяемая рекуррентными соотношениями

$$W_N^k = \left(W_N^{k-1} + \int_{t_{N-k}}^{t_{N-k+1}} \pi e^{(T-s)A} B P ds \right) * \int_{t_{N-k}}^{t_{N-k+1}} \pi e^{(T-s)A} C Q ds, \quad k = 1, \dots, N,$$

$$W_N^0 = M.$$

Из свойств операций над множествами следует, что альтернированная сумма W_N^N является выпуклым компактным множеством. Кроме того справедлива следующая лемма.

Лемма 1. Пусть ω_{N_1} и ω_{N_2} – два произвольных разбиения отрезка $[0, T]$, причем $\omega_{N_2} \subset \omega_{N_1}$, то есть первое разбиение содержит все точки второго. Тогда

$$W_{N_1}^{N_1} \subset W_{N_2}^{N_2}.$$

Определение 14. Множество

$$W = \bigcap_N W_N^N, \quad (5)$$

где пересечение берется по всевозможным разбиениям ω_N отрезка $[0, T]$, называется альтернированным интегралом Л.С. Понтрягина.

Из теории дифференциальных игр [1, 5, 9, 8] следует следующая теорема

Теорема 2. Пусть альтернированный интеграл $W \neq \emptyset$ и выполняется условие $z(0) \in W$. Тогда задача наведения из позиции $(0, x_0)$ на терминальное множество M в момент времени T разрешима в классе позиционных стратегий первого игрока.

В противном случае в классе позиционных стратегий второго игрока разрешима задача уклонения из позиции $(0, x_0)$ от терминального множества M в момент времени T .

Таким образом, условие

$$z(0) \in W$$

является достаточным условием решения задачи наведения из начальной позиции $(0, x_0)$ на терминальное множество M в момент времени T .

Достаточным условием решения задачи уклонения из той же позиции является обратное условие

$$z(0) \notin W.$$

4 Метод решения

Практическое вычисление альтернированного интеграла наталкивается на значительные трудности, связанные в частности с вычислением бесконечного пересечения в формуле (5). Из леммы 1 следует, что при достаточно мелком разбиении ω_N отрезка $[0, T]$ альтернированная сумма W_N^N является "хорошей" аппроксимацией множества W . В дальнейшем будем строить именно альтернированные суммы.

Рассмотрим некоторое равномерное разбиение ω_N отрезка $[0, T]$ такое, что диаметр разбиения равен $h = \frac{T}{N}$, а точки $t_i = ih$, $i = 0, \dots, N$. Будем предполагать, что игроки пользуются кусочно постоянными стратегиями и могут изменять свое управление только в моменты времени из разбиения ω_N , оставляя его постоянным между этими моментами. Рекуррентная формула для вычисления альтернированных сумм в этом случае примет вид

$$W_N^k = \left(W_N^{k-1} + \pi \int_{t_{N-k}}^{t_{N-k+1}} e^{(T-s)A} ds BP \right) * \\ * \pi \int_{t_{N-k}}^{t_{N-k+1}} e^{(T-s)A} ds CQ, \quad k = 1, \dots, N.$$

Здесь вместо интеграла от многозначного отображения мы имеем интеграл от матричной функции. После дальнейших упрощений получаем

$$W_N^k = \left(W_N^{k-1} + \pi e^{(T-t_{N-k})A} \int_0^{t_{N-k+1}-t_{N-k}} e^{-\tau A} d\tau BP \right) * \\ * \pi e^{(T-t_{N-k})A} \int_0^{t_{N-k+1}-t_{N-k}} e^{-\tau A} d\tau CQ, \quad k = 1, \dots, N. \quad (6)$$

Введем следующие обозначения

$$A_k = e^{khA} \int_0^h e^{-\tau A} d\tau \subset R^{n \times n}, \quad k = 1, \dots, N,$$

$$P_k = \pi A_k BP, \quad Q_k = \pi A_k CQ \subset R^\nu,$$

В новых обозначениях формула (6) примет вид

$$W_N^0 = M, \quad (7)$$

$$W_N^k = (W_N^{k-1} + P_k) * Q_k, \quad k = 1, \dots, N. \quad (8)$$

Достаточным условием для решения задачи наведения является выполнение включения

$$\pi e^{TA} x_0 \in W_N^N. \quad (9)$$

Таким образом, для вычисления гарантированного времени решения задачи наведения необходимо проверить условие (9) для всех конечных моментов времени $T \in [0, \Theta]$.

Рассмотрим одно свойство альтернированных сумм, позволяющее не пересчитывать их заново для нового момента времени. Пусть задан произвольный момент времени $T \in [0, \Theta]$ и равномерное разбиение ω_N отрезка $[0, T]$ с диаметром h . Рассмотрим отрезок $[0, T + h]$ и его равномерное разбиение ω_{N+1} с тем же диаметром. Заметим, что для альтернированных сумм будет выполняться равенство

$$W_N^i = W_{N+1}^i, \quad i = 0, \dots, N.$$

То есть для вычисления альтернированной суммы W_{N+1}^{N+1} , зная множество W_N^N , необходимо сделать только один шаг по формуле (8) для $k = N + 1$.

Приходим к следующему алгоритму вычисления гарантированного времени решения задачи наведения.

1. По выбранному числу N строим разбиение ω_N всего отрезка $[0, \Theta]$.
2. Последовательно строим множества W_N^k , $k = 0, \dots, N$, используя рекуррентные соотношения (7), (8), и проверяем включение

$$\pi e^{khA} x_0 \in W_N^k. \quad (10)$$

Первый индекс k , для которого выполняется это включение, обозначим буквой \bar{k} . Тогда время, за которое задача наведения будет *гарантированно* разрешена, равно

$$T = \bar{k}h.$$

Если при каком-либо k множество W_N^k оказалось пустым или условие (10) не выполняется для всех $k = 0, \dots, N$, то задача наведения считается неразрешимой и соответственно *разрешима задача уклонения*.

Заметим, что данный алгоритм дает приближенное значение гарантированного времени, так как достаточное условие разрешимости задачи наведения проверяется только в точках t_k из разбиения отрезка $[0, \Theta]$. Точность вычисления гарантированного времени зависит от выбора числа N точек разбиения ω_N и от точности построения альтернированных сумм. Оценки для погрешности вычислений можно найти в работе [9].

4.1 Вычисление альтернированной суммы

Как было показано ранее, все множества, используемые в формулах (7), (8), являются выпуклыми компактами. Следовательно, для их описания удобно пользоваться опорными функциями.

Определение 15. Пусть $X \subset R^n$ – замкнутое выпуклое множество. Его опорной функцией называется функция $c(X, \psi)$ от произвольного вектора $\psi \in R^n$, определяемая формулой

$$c(X, \psi) = \sup_{x \in X} \langle x, \psi \rangle.$$

Так как разбиение отрезка $[0, \Theta]$ зафиксировано, то не будем далее отображать зависимость множеств W_N^k от разбиения, то есть будем их обозначать просто W^k . В терминах опорных функций формулы (7), (8) запишутся следующим образом

$$c(W^0, \psi) = c(M, \psi), \quad (11)$$

$$c(W^k, \psi) = \text{co}(c(W^{k-1}, \psi) + c(P_k, \psi) - c(Q_k, \psi)), \quad k = 1, \dots, N, \quad (12)$$

где через $\text{co}(c(\cdot))$ обозначена выпуклая оболочка функции $c(\cdot)$, а опорные функции множеств P , Q и M считаются заданными.

В силу необходимости приближенного вычисления опорных функций при практической реализации необходимо ограничить число точек, в которых эти функции будут вычисляться. Из положительной однородности опорной функции такое конечное множество точек можно выбирать на единичной сфере. При этом такое множество должно быть достаточно плотным на единичной сфере, чтобы вычислять альтернированные суммы с достаточной точностью.

Рассмотрим в пространстве R^ν сетку на единичной сфере, состоящую из m точек

$$\Psi = \{\psi_1, \psi_2, \dots, \psi_m\}.$$

Описание методов построения таких сеток приведено в разделе 6.

Переходим к алгоритму приближенного построения опорной функции альтернированной суммы.

Шаг 0. На начальном этапе для всех точек из сетки Ψ вычисляется значение опорной функции множества W^0 по формулам

$$c(W^0, \psi_i) = c(M, \psi_i), \quad i = 1, \dots, m.$$

.....

Пусть теперь на $k - 1$ -м шаге ($k \in \{1, \dots, N\}$) вычислена опорная функция $c(W^{k-1}, \psi)$ для всех $\psi \in \Psi$.

Шаг k . Вычисляется значение опорной функции множества W^k для всех точек сетки Ψ по формулам

$$c(W^k, \psi_i) = \text{co}(c(W^{k-1}, \psi_i) + c(P_k, \psi_i) - c(Q_k, \psi_i)), \quad i = 1, \dots, m,$$

где

$$c(P_k, \psi_i) = c(P, (\pi A_k B)^T \psi_i),$$

$$c(Q_k, \psi_i) = c(Q, (\pi A_k C)^T \psi_i),$$

а алгоритм выпукления положительно однородной функции приведен в разделе 5.

После выполнения шага N имеем опорные функции альтернированных сумм для всех моментов времени из разбиения ω_N .

4.2 Гарантированное время окончания преследования

Перепишем условие (10), используя опорные функции, в виде

$$c(W^k, \psi) \geq \langle \pi e^{khA} x_0, \psi \rangle, \quad \forall \psi \in S_1.$$

Тогда проверку условия (10) можно заменить проверкой условия

$$\min_{\psi \in \Psi} \{c(W^k, \psi) - \langle \pi e^{khA} x_0, \psi \rangle\} \geq 0. \quad (13)$$

Таким образом, индекс \bar{k} — это первый индекс, для которого выполняется неравенство

$$\min_{\psi \in \Psi} \{c(W^{\bar{k}}, \psi) - \langle \pi e^{\bar{k}hA} x_0, \psi \rangle\} \geq 0, \quad (14)$$

а $T = \bar{k}h$ — гарантированное время решения задачи наведения.

4.3 Позиционные стратегии преследования и уклонения

Пусть вычислено гарантированное время окончания процесса преследования T , и это время имеет вид $T = \bar{k}h$, то есть точка $\pi e^{\bar{k}hA} x_0$ принадлежит альтернированной сумме $W^{\bar{k}}$.

Обозначим через $\tilde{\psi}_{\bar{k}}(x_0)$ любой вектор, на котором достигается минимум в (14), то есть

$$\tilde{\psi}_{\bar{k}}(x_0) \in \operatorname{argmin}_{\psi \in \Psi} \left\{ c(W^{\bar{k}}, \psi) - \langle \pi e^{\bar{k}hA} x_0, \psi \rangle \right\}. \quad (15)$$

Тогда *позиционная стратегия преследования* в момент времени $t = 0$ выбирается по правилу

$$U(0, x_0) \in \operatorname{argmax}_{u \in P} \langle \pi e^{\bar{k}hA} B u, \tilde{\psi}_{\bar{k}}(x_0) \rangle, \quad (16)$$

позиционная стратегия уклонения в момент времени $t = 0$ выбирается по правилу

$$V(0, x_0) = \operatorname{argmax}_{v \in Q} \langle \pi e^{\bar{k}hA} C v, \tilde{\psi}_{\bar{k}}(x_0) \rangle. \quad (17)$$

Рассмотрим геометрический смысл формул (16), (17). Выполнение условия (14) означает, что точка $\pi e^{\bar{k}hA} x_0$ лежит на границе $W^{\bar{k}}$, так как \bar{k} — первый индекс, когда это условие выполняется. Тогда вектор $\tilde{\psi}_{\bar{k}}(x_0)$ будет опорным вектором ко множеству $W^{\bar{k}}$ в точке $\pi e^{\bar{k}hA} x_0$.

Первый игрок не знает, какое управление будет выбрано вторым игроком, но его цель заключается в том, чтобы удержать траекторию на следующем шаге на множестве $W^{\bar{k}-1}$. Поэтому рациональным поведением для него является так выбирать свое управление, чтобы вектор скорости динамической системы (4) был направлен противоположно направлению вектора $\tilde{\psi}_{\bar{k}}(x_0)$, то есть направлен ко множеству $W^{\bar{k}}$. Следовательно, управление первого игрока нужно выбирать по правилу

$$\begin{aligned} U(0, x_0) &\in \operatorname{argmin}_{u \in P} \langle -\pi e^{(T-t)A} B u(t) + \pi e^{(T-t)A} C v(t), \tilde{\psi}_{\bar{k}}(x_0) \rangle = \\ &= \operatorname{argmax}_{u \in P} \langle \pi e^{\bar{k}hA} B u, \tilde{\psi}_{\bar{k}}(x_0) \rangle. \end{aligned}$$

Второй игрок, наоборот, пытается максимально удалиться на следующем шаге от множества $W^{\bar{k}-1}$. Поэтому должен выбирать свое управление таким образом, чтобы вектор скорости был сонаправлен с вектором $\tilde{\psi}_{\bar{k}}(x_0)$, то есть направлен от множества $W^{\bar{k}}$. Следовательно, управление второго игрока нужно выбирать по правилу

$$\begin{aligned} V(0, x_0) &\in \operatorname{argmax}_{v \in Q} \langle -\pi e^{(T-t)A} B u(t) + \pi e^{(T-t)A} C v(t), \tilde{\psi}_{\bar{k}}(x_0) \rangle = \\ &= \operatorname{argmax}_{v \in Q} \langle \pi e^{\bar{k}hA} C v, \tilde{\psi}_{\bar{k}}(x_0) \rangle. \end{aligned}$$

После того как выбраны управления $U(0, x_0)$ и $V(0, x_0)$, подставим их в систему (1) и предполагая, что управления постоянны на отрезке времени $[0, h]$, проинтегрируем эту систему, чтобы получить значения траектории $x(h)$.

Позиционные стратегии первого и второго игроков на следующих шагах выбирается из тех же соображений, что и на первом шаге. Запишем общую формулу вычисления стратегий для $k = \bar{k}, \bar{k} - 1, \dots, 1$

$$\tilde{\psi}_k(x(T - kh)) \in \operatorname{argmin}_{\psi \in \Psi} \{c(W^k, \psi) - \langle \pi e^{khA} x(T - kh), \psi \rangle\},$$

$$U(T - kh, x(T - kh)) \in \operatorname{argmax}_{u \in P} \langle \pi e^{khA} B u, \tilde{\psi}_k(x(T - kh)) \rangle, \quad (18)$$

$$V(T - kh, x(T - kh)) \in \operatorname{argmax}_{v \in Q} \langle \pi e^{khA} C v, \tilde{\psi}_k(x(T - kh)) \rangle. \quad (19)$$

После того как выбраны управления $U(T - kh, x(T - kh))$ и $V(T - kh, x(T - kh))$, интегрируем систему (1) на отрезке времени $[T - kh, T - (k - 1)h]$, чтобы получить значение траектории $x(T - (k - 1)h)$.

Геометрический смысл вектора $\tilde{\psi}_k(x(T - kh))$ следующий.

1. если точка $\pi e^{khA} x(T - kh)$ лежит на границе множества W^k , то вектор $\tilde{\psi}_k(x(T - kh))$ является опорным вектором к выпуклому множеству W^k ;

2. если точка $\pi e^{khA} x(T - kh)$ лежит вне множества W^k , то вектор $\tilde{\psi}_k(x(T - kh))$ сонаправлен с вектором, соединяющим точку $\pi e^{khA} x(T - kh)$ и точку ее проекции на множество W^k и направленным от множества W^k .

Тогда первый игрок на шаге k выбирает свое управление (18) так, чтобы траектория системы на следующем шаге как можно ближе находилась от множества W^{k-1} , а второй игрок выбирает свое управление (19) так, чтобы траектория системы на следующем шаге как можно дальше находилась от множества W^{k-1} .

В теории позиционных дифференциальных игр, такой способ построения управления преследования называется *экстремальным сдвигом* [8], [10].

5 Вычисление геометрической разности двух выпуклых компактов

Рассмотрим два выпуклых компакта A и B в пространстве R^n и их геометрическую разность

$$D = A * B. \quad (20)$$

Если множество D не пусто, то оно является выпуклым компактом.

Опорная функция $c(D, \psi)$ множества D определяется через опорные функции $c(A, \psi)$, $c(B, \psi)$ множеств A и B равенством

$$c(D, \psi) = \text{co}(c(A, \cdot) - c(B, \cdot))(\psi). \quad (21)$$

Таким образом, вычисление опорной функции геометрической разности двух компактных множеств сводится к овыпуклению положительно однородной функции

$$c(\psi) = c(A, \psi) - c(B, \psi).$$

В общем случае всегда выполняется неравенство

$$c(D, \psi) \leq c(A, \psi) - c(B, \psi), \quad \psi \in R^n.$$

Рассмотрим пример необходимости применения операции овыпукления.

Пример. Рассмотрим два множества

$$A = \{x \in R^2 : |x_1| \leq 1, |x_2| \leq 1\},$$

$$B = S_{0,9}(0, 0).$$

Очевидно

$$D = A * B = \{x \in R^2 : |x_1| \leq 0.1, |x_2| \leq 0.1\}.$$

Опорные функции имеют вид

$$c(D, \psi) = 0.1(|\psi_1| + |\psi_2|),$$

$$c(\psi) = c(A, \psi) - c(B, \psi) = |\psi_1| + |\psi_2| - 0.9\sqrt{\psi_1^2 + \psi_2^2}.$$

На рисунках 1, 2 приведены графики этих функций. Нетрудно заметить, что функция $c(\psi)$ не является выпуклой и не совпадает с опорной функцией геометрической разности $c(D, \psi)$.

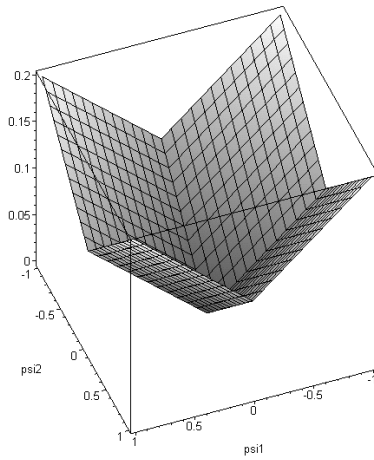


Рис. 1. График опорной функции геометрической разности $c(D, \psi)$.

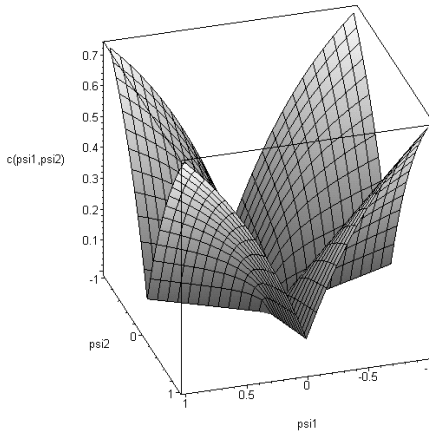


Рис. 2. График функции $c(\psi) = c(A, \psi) - c(B, \psi)$.

5.1 Вычисление выпуклой оболочки положительно однородной функции

Выпуклой оболочкой $\text{co}(f(\cdot))(\psi)$ функции $f(\psi)$, где $\psi \in R^n$, называется наибольшая выпуклая функция, не превосходящая $f(\psi)$ для всех $\psi \in R^n$. Функция $\text{co}(f(\cdot))(\psi)$ допускает представление

$$\text{co}(f(\cdot))(\psi) = \inf_{\{\lambda_i, \psi_i: \sum_{i=1}^n \lambda_i \psi_i = \psi, \sum_{i=1}^n \lambda_i = 1, \lambda_i \geq 0\}} \sum_{i=1}^n \lambda_i f(\psi_i).$$

Рассмотрим в пространстве R^n положительно однородную функцию $c(\psi)$ и сетку на единичной сфере $\Psi = \{\psi_1, \psi_2, \dots, \psi_m\}$. Считаем, что функция $c(\psi)$ задана на векторах сетки Ψ , то есть известны значения

$$c(\psi_i), \quad i = 1, \dots, m.$$

Будем искать приближенные значения выпуклой оболочки функции $c(\psi)$ на векторах сетки Ψ . Эти значения обозначим

$$\tilde{c}_{\Psi} c(\psi_i), \quad i = 1, \dots, m.$$

Выпуклую положительно однородную функцию можно рассматривать как опорную функцию выпуклого множества. Обозначим

$$F = \bigcap_{\psi \in \Psi} \{f \in R^n : \langle f, \psi \rangle \leq \tilde{c}_{\Psi} c(\psi)\}, \quad (22)$$

то есть множество, которое соответствует функции $\tilde{c}_{\Psi} c(\psi)$.

Центральным является следующее утверждение.

Лемма 2. *Для множества F , определяемого равенством (22), верно соотношение*

$$F = \bigcap_{\psi \in \Psi} \{f \in R^n : \langle f, \psi \rangle \leq c(\psi)\}. \quad (23)$$

Таким образом, если построить множество F по формуле (23) и найти его опорную функцию, то эта опорная функция и будет выпуклой оболочкой положительно однородной функции $c(\psi)$. Отметим, что в формуле (22) функцию $c(\psi)$ нельзя считать опорной функцией множества F , так как $c(\psi)$ невыпуклая.

Техническая часть алгоритма базируется на следующей лемме.

Лемма 3. Пусть заданы два единичных вектора ξ , ψ и $\langle \psi, \xi \rangle > 0$. Тогда, точка пересечения луча $\cup_{\lambda \geq 0} \{\lambda \xi\}$ с гиперплоскостью $\langle x, \psi \rangle = c$ задается формулой

$$\lambda = \frac{c}{\langle \psi, \xi \rangle}. \quad (24)$$

Из этой леммы следует формула: Пусть внутренность множества F содержит нулевой вектор. Вычислим число

$$\lambda(\xi) = \min \left\{ \frac{c(\psi)}{\langle \psi, \xi \rangle} : \psi \in \Psi, \langle \psi, \xi \rangle > 0 \right\}, \quad (25)$$

тогда вектор $\lambda(\xi)\xi$ приближенно лежит на пересечении границы множества F с лучом $\cup_{\lambda \geq 0} \{\lambda \xi\}$.

Формула (25) верна, если внутренность множества F содержит нулевой вектор, или в терминах опорных функций

$$\min_{i=1, \dots, m} c(\psi_i) > 0.$$

Рассмотрим алгоритм выпукления положительно однородной функции. Пусть задана сетка Ψ на единичной сфере и положительно однородная функция $c(\psi)$. Найдем значения функции $\tilde{c}_\Psi c(\psi_i)$ на векторах сетке Ψ . Для этого

1. Для каждого вектора $\psi_i \in \Psi$, $i = 1, \dots, m$, вычислим

$$\lambda(\psi_i) = \min_{k=1, \dots, m} \left\{ \frac{c(\psi_k)}{\langle \psi_k, \psi_i \rangle} : \langle \psi_k, \psi_i \rangle > 0 \right\}, \quad i = 1, \dots, m.$$

Обозначим через Ψ_1 множество векторов ψ_k сетки Ψ , соответствующих тем индексам k , на которых хотя бы однажды достигался минимум при вычислении чисел $\lambda(\psi_i)$.

2. Положим $\Psi_2 = \Psi \setminus \Psi_1$. Аппроксимация выпуклой оболочки функции $c(\psi)$ вычисляется по следующей формуле

$$\tilde{c}_\Psi c(\psi_i) = \begin{cases} c(\psi_i), & \psi_i \in \Psi_1, \\ \max_{k=1, \dots, m} \langle \lambda(\psi_k) \psi_k, \psi_i \rangle, & \psi_i \in \Psi_2. \end{cases}$$

Алгоритм применим, если $\min_{i=1, \dots, m} c(\psi_i) > 0$ (то есть внутренность множества F содержит нулевой вектор).

В случае, когда это условие не выполнено, вычислим функцию $c_s(\psi)$, которая соответствует множеству F_s , получаемого сдвигом множества

F на некоторый вектор f_s таким образом, чтобы оно содержало нулевой вектор. Функция $c_s(\psi)$ будет определяться равенством

$$c_s(\psi) = c(\psi) - \langle f_s, \psi \rangle.$$

Затем применяем алгоритм овыпукления к функции $c_s(\psi)$, получаем функцию $\text{co}_\Psi c_s(\psi)$ и делаем обратный сдвиг

$$\tilde{\text{co}}_\Psi c(\psi) = \tilde{\text{co}}_\Psi c_s(\psi) + \langle f_s, \psi \rangle.$$

Определение 16. Центром Штейнера выпуклого компакта $F \in R^n$ называется точка

$$\bar{w}(F) = \frac{1}{v_n} \int_S c(F, \psi) \psi dS,$$

где v_n — объем единичного шара в пространстве R^n .

Вычислим приближенно вектор $\bar{w}(F)$, используя только вектора сетки Ψ . Воспользуемся равенством

$$v_n = \frac{S_n}{n},$$

где S_n — площадь поверхности единичного шара в пространстве R^n . Тогда

$$\begin{aligned} \frac{1}{v_n} \int_S c(F, \psi) \psi dS &\approx \frac{n}{S_n} \sum_{i=1}^m c(F, \psi_i) \psi_i \Delta S_n = \\ &= \frac{n}{m \Delta S_n} \sum_{i=1}^m c(F, \psi_i) \psi_i \Delta S_n = \\ &= \frac{n}{m} \sum_{i=1}^m c(F, \psi_i) \psi_i. \end{aligned}$$

В качестве вектора сдвига множества F можно взять вектор

$$f_s = \frac{n}{m} \sum_{i=1}^m c(\psi_i) \psi_i,$$

приблизительно совпадающий с центром Штейнера множества F .

5.2 Алгоритм приближенного вычисления геометрической разности

Рассмотрим алгоритм приближенного вычисления опорной функции множества D .

Пусть на единичной сфере в пространстве R^n задана сетка

$$\Psi = \{\psi_1, \psi_2, \dots, \psi_m\}, \quad (26)$$

содержащая m векторов. Будем считать, что любой вектор пространства R^n может быть представлен в виде линейной комбинации векторов из сетки Ψ с неотрицательными коэффициентами.

Будем искать значения опорной функции $c(D, \psi)$ на векторах сетки Ψ , то есть значения

$$c(D, \psi_i), \quad i = 1, \dots, m.$$

Обозначим

$$c_i = c(A, \psi_i) - c(B, \psi_i), \quad i = 1, \dots, m.$$

Вычислим вектор $d_c \in R^n$, определяемый формулой

$$d_c = \frac{n}{m} \sum_{i=1}^m c_i \psi_i.$$

Обозначим

$$\tilde{c}_i = c_i - \langle d_c, \psi_i \rangle, \quad i = 1, \dots, m.$$

Вычислим числа

$$\lambda_i = \min_{k=1, \dots, m} \left\{ \frac{\tilde{c}_k}{\langle \psi_k, \psi_i \rangle} : \langle \psi_k, \psi_i \rangle > 0 \right\}, \quad i = 1, \dots, m. \quad (27)$$

Обозначим через Ψ_1 множество векторов ψ_k сетки Ψ , соответствующих тем индексам k , на которых хотя бы однажды достигался минимум при вычислении чисел λ_i по формуле (27). И обозначим через Ψ_2 множество $\Psi \setminus \Psi_1$.

Искомые значения опорной функции $c(D, \psi)$ на сетке Ψ определяются следующим образом

$$c(D, \psi_i) = \begin{cases} c_i, & \psi_i \in \Psi_1, \\ \max_{k=1, \dots, m} \langle \lambda_k \psi_k, \psi_i \rangle + \langle d_c, \psi_i \rangle, & \psi_i \in \Psi_2. \end{cases} \quad (28)$$

Формула (28) дает приближенное значение опорной функции на сетке Ψ .

5.3 Визуализация геометрической разности и проверка на пустоту

Важное свойство чисел λ_i , $i = 1, \dots, m$, определяемых формулой (27), заключается в том, что векторы $d_c + \lambda_i \psi_i$, $i = 1, \dots, m$ приближенно лежат на границе множества D . Поэтому, эти векторы можно использовать для приближенного построения границы множества D , а само множество отождествить с выпуклым многогранником

$$D \approx \text{co}\{d_c + \lambda_1 \psi_1, \dots, d_c + \lambda_m \psi_m\}.$$

Заметим, что множество D в случае его непустоты по построению должно содержать в себе точку d_c . То есть все числа λ_i , $i = 1, \dots, m$ должны быть неотрицательными. Если это условие не выполняется, то считаем, что геометрическая разность является пустым множеством.

6 Алгоритмы построения многомерных сеток

В данном разделе рассматриваются следующие вопросы:

1. Построение сетки на поверхности единичной сферы для 2-х мерного случая.
2. Построение сетки по гиперкубу с узлами в целочисленных координатах в общем случае. Под гиперкубом понимается куб в пространстве произвольной размерности R^n .
3. Переход от сетки по гиперкубу с узлами в целочисленных координатах к сетке гиперкуба с координатами узлов из отрезка $[-1, 1]$.
4. Построение сетки по поверхности гиперкуба с координатами узлов из отрезка $[-1, 1]$.
5. Преобразование сетки по поверхности гиперкуба в сетку по поверхности единичной гиперсферы.

Параметры алгоритма:

- Входные данные
 - n – размерность пространства, в котором строится сетка;

– m – число разбиений одного измерения гиперкуба.

- Выходные данные

- $\Psi \subset R^n$ – конечное множество узлов сетки на поверхности гиперсферы единичного радиуса.

6.1 Построение сетки на поверхности единичной сферы для 2-х мерного случая

Сфера размерности два – это окружность. В полярных координатах все точки окружности отличаются только углом $\varphi \in [0, 2\pi)$. Соответственно равномерная сетка из m точек на единичной окружности может быть описана следующим образом.

$$\Psi = \left\{ \left(\cos \frac{2\pi i}{m}, \sin \frac{2\pi i}{m} \right) : i = 0, \dots, m - 1 \right\}.$$

6.2 Построение сетки на поверхности единичной гиперсферы в общем случае

В случае произвольной размерности не удастся сразу задать сетку на поверхности единичной гиперсферы, то есть сферы в пространстве R^n . Поэтому рассматривается более простая задача – построение сетки по всему гиперкубу. Далее из этой сетки удаляются внутренние точки и осуществляется ее отображение на поверхность единичной гиперсферы.

Сетка по всему гиперкубу с узлами в целочисленных координатах задается следующим образом.

$$\Psi_m = \{(i_1, i_2, \dots, i_n) : i_j = 0, \dots, m, j = 1, \dots, n\}.$$

Проблема при практическом построении такой сетки заключается в том, что размерность n заранее не известна, и соответственно не известно количество вложенных циклов по переменным i_j .

Один из способов решения данной проблемы состоит с отождествлением каждого элемента из множества Ψ_m с некоторым числом в $(m+1)$ -значной системе исчисления. То есть каждому вектору (i_1, i_2, \dots, i_n) поставим в соответствие число $\overline{i_1, i_2, \dots, i_n}$. Всего таких чисел и, соответственно, узлов сетки $(m+1)^n$.

Для получения всех узлов сетки начинаем с числа 0, которому соответствует вектор $(0, 0, \dots, 0)$. Далее прибавляем на каждом шаге к

нашему числу единицу и получаем новый узел сетки. Повторяя эту операцию $(m + 1)^n$ раз, получаем все узлы искомой сетки Ψ_m .

Далее совершим переход от сетки по гиперкубу с узлами в целочисленных координатах к сетке гиперкуба с координатами узлов из отрезка $[-1, 1]$. Для этого применим сдвиг и сжатие векторов и получим новую сетку Ψ_1 :

$$\Psi_1 = \left\{ \left(\frac{2\psi_1}{m} - 1, \frac{2\psi_2}{m} - 1, \dots, \frac{2\psi_n}{m} - 1 \right) : (\psi_1, \psi_2, \dots, \psi_n) \in \Psi_m \right\}$$

Рассмотрим теперь вопрос построения сетки по поверхности гиперкуба с координатами узлов из отрезка $[-1, 1]$. Для этого необходимо исключить из множества Ψ_1 все узлы, не лежащие на границе гиперкуба. Чем отличаются точки грани от прочих точек гиперкуба? Тем, что у них одна из координат зафиксирована в крайнем положении, то есть равняется ± 1 . Таким образом отбрасывая узлы, у которых все координаты лежат внутри отрезка $[-1, 1]$ получаем сетку по поверхности гиперкуба

$$\partial\Psi_1 = \{(\psi_1, \psi_2, \dots, \psi_n) \in \Psi_1 : \exists \psi_i = \pm 1\}.$$

Переход от сетки по поверхности гиперкуба к искомой сетке по поверхности гиперсферы состоит в нормировке векторов сетки

$$\Psi = \left\{ \frac{\psi}{\|\psi\|} : \psi \in \partial\Psi_1 \right\}.$$

7 Примеры для тестирования программы

В этом разделе собраны примеры, в которых имеется аналитическая формула для вычисления альтернированного интеграла и гарантированного времени решения задачи наведения [1].

В предлагаемых примерах все множества, участвующие в построении альтернированной суммы (см. формулы (7),(8)), являются шарами соответствующего радиуса с центром в начале координат. Из свойств геометрической разности следует, что для вычисления геометрической разности двух шаров, операции вычисления выпуклой оболочки функции не требуется. Данный факт может облегчить отладку программы.

7.1 Два Крокодила R^2

$$\begin{cases} \dot{z}_1 = z_2, & z_1(0) = z_1^0, \\ \dot{z}_2 = u - v, & z_2(0) = z_2^0. \end{cases}$$

$$z_i \in R^2, \quad i = 1, 2, \quad \|u\| \leq \rho, \quad \|v\| \leq \sigma, \quad M = \{z : \|z_1\| \leq l\}$$

Гарантированное время окончания игры преследования из позиции $(0, z_1^0, z_2^0)$ ищется, как наименьший положительный корень уравнения:

$$\left(l + (\rho - \sigma) \frac{t^2}{2} \right)^2 = \|z_1^0\|^2 + 2t \langle z_1^0, z_2^0 \rangle + t^2 \|z_2^0\|^2.$$

В следующей таблице приведены значения гарантированного времени решения задачи преследования при разных начальных позициях и различных ресурсах управлений игроков.

l	ρ	σ	z_{11}^0	z_{12}^0	z_{21}^0	z_{22}^0	T
1	2	1	0	7	-1	1	4.862
1	2	1	2	1	-1	1	2
1	2	1	2	-1	-1	1	0.7466
1	2	1	3	-1	-1	1	1.24357
1	2	1	3	-1	-2	1	0.8321
1	2	1	3	-1	-2	-2	4.78369
1	1.5	1	-3	-1	-2	-2	11.9356
1	1.5	1	-1	2	-2	-2	10.70418
1	1.5	1	-1	2	1	-2	0.52
1	1.5	1	-1	2	1	-1	0.83
1	1.5	1	-1	-2	1	-1	5.63

7.2 Контрольный пример Понтрягина R^2

$$\begin{cases} \dot{z}_1 = z_2 - z_3, & z_1(0) = z_1^0, \\ \dot{z}_2 = -\alpha z_2 + u, & z_2(0) = z_2^0, \\ \dot{z}_3 = -\beta z_3 + v, & z_3(0) = z_3^0. \end{cases}$$

$$z_i \in R^2, \quad i = 1, 2, 3, \quad \|u\| \leq \rho, \quad \|v\| \leq \sigma, \quad M = \{z : \|z_1\| = 0\}$$

Гарантированное время окончания игры преследования из позиции $(0, z_1^0, z_2^0, z_3^0)$ ищется, как наименьший положительный корень уравнения:

$$\left\| z_1^0 + \frac{1 - e^{-\alpha t}}{\alpha} z_2^0 - \frac{1 - e^{-\beta t}}{\beta} z_3^0 \right\| = \frac{\rho}{\alpha} \left(t + \frac{e^{-\alpha t} - 1}{\alpha} \right) - \frac{\sigma}{\beta} \left(t + \frac{e^{-\beta t} - 1}{\beta} \right).$$

В следующей таблице приведены значения гарантированного времени решения задачи преследования при разных начальных позициях и различных ресурсах управлений игроков.

α	β	ρ	σ	z_{11}^0	z_{12}^0	z_{21}^0	z_{22}^0	z_{31}^0	z_{32}^0	T
3	1	4	1	2	-1	-2	1	-1	1	6.7934
3	1	4	1	2	-2	-2	1	-1	1	8.9
3	1	4	1	-1	-2	-2	1	-1	1	7.27
3	1	4	1	1	-2	0	1	0	1	6.877
3	1	4	1	1	-2	0	0	0	0	5.06056
3	1	4	1	1	-1	0	0	0	0	2.764
2	1	4	1	1	-1	0	0	0	0	1.578
2	1	4	1	2	-1	0	0	0	0	2.3243
2	1	4	1	2	-1	2	-1	0	0	3.3855

7.3 Два Крокодила R^3

$$\begin{cases} \dot{z}_1 = z_2, & z_1(0) = z_1^0, \\ \dot{z}_2 = u - v, & z_2(0) = z_2^0. \end{cases}$$

$$z_i \in R^3, \quad i = 1, 2, \quad \|u\| \leq \rho, \quad \|v\| \leq \sigma, \quad M = \{z : \|z_1\| \leq l\}$$

Гарантированное время окончания игры преследования из позиции $(0, z_1^0, z_2^0)$ ищется, как наименьший положительный корень уравнения:

$$\left(l + (\rho - \sigma) \frac{t^2}{2} \right)^2 = \|z_1^0\|^2 + 2t \langle z_1^0, z_2^0 \rangle + t^2 \|z_2^0\|^2.$$

В следующей таблице приведены значения гарантированного времени решения задачи преследования при разных начальных позициях и различных ресурсах управлений игроков.

1	ρ	σ	z_{11}^0	z_{12}^0	z_{13}^0	z_{21}^0	z_{22}^0	z_{23}^0	T
1	1.5	1	-1	-2	1	1	-1	2	9.924
1	3	1	-1	-2	1	1	-1	2	2.643
1	3	1	2	-2	1	1	-1	0	2.2598
1	3	1	2	0	1	1	-1	0	1.835
1	2	1	2	0	1	1	-1	-1	3.236
2	2	1	3	0	1	1	-1	-1	3.3172

7.4 Контрольный пример Понтрягина R^3

$$\begin{cases} \dot{z}_1 = z_2 - z_3, & z_1(0) = z_1^0, \\ \dot{z}_2 = -\alpha z_2 + u, & z_2(0) = z_2^0, \\ \dot{z}_3 = -\beta z_3 + v, & z_3(0) = z_3^0. \end{cases}$$

$$z_i \in R^3, \quad i = 1, 2, 3, \quad \|u\| \leq \rho, \quad \|v\| \leq \sigma, \quad M = \{z : \|z_1\| = 0\}$$

Гарантированное время окончания игры преследования из позиции $(0, z_1^0, z_2^0, z_3^0)$ ищется, как наименьший положительный корень уравнения:

$$\begin{aligned} \left\| z_1^0 + \frac{1 - e^{-\alpha t}}{\alpha} z_2^0 - \frac{1 - e^{-\beta t}}{\beta} z_3^0 \right\| = \\ = \frac{\rho}{\alpha} \left(t + \frac{e^{-\alpha t} - 1}{\alpha} \right) - \frac{\sigma}{\beta} \left(t + \frac{e^{-\beta t} - 1}{\beta} \right). \end{aligned}$$

В следующей таблице приведены значения гарантированного времени решения задачи преследования при разных начальных позициях и различных ресурсах управлений игроков.

α	β	ρ	σ	z_{11}^0	z_{12}^0	z_{13}^0	z_{21}^0	z_{22}^0	z_{23}^0	z_{31}^0	z_{32}^0	z_{33}^0	T
2	1	4	1	2	-1	1	2	-1	1	0	0	1	3.426
2	1	4	1	2	-2	1	2	-1	1	0	0	1	3.957
2	1	4	1	2	-2	1	0	-1	1	0	0	1	3.28
2	1	4	1	2	-2	1	0	-1	1	-1	0	1	3.94
2	1	4	1	2	-2	1	1	-1	1	-1	-1	1	3.852
2	1	4	1	1	-1	1	0	-1	1	0	0	1	2.622

8 Опорные функции некоторых множеств в R^n

Сфера $S_\rho^n(a) = \{x \in R^n : \|x - a\| \leq \rho\}$,

$$c(S_\rho^n(0), \psi) = (a, \psi) + \rho \|\psi\| = a_1 \psi_1 + \dots + a_n \psi_n + \rho \sqrt{\psi_1^2 + \dots + \psi_n^2}.$$

Параллелепипед $P^n(a) = \{x \in R^n : -a_i \leq x_i \leq a_i, i = 1, \dots, n\}$,

$$c(P^n(a), \psi) = \sum_{i=1}^n a_i |\psi_i|.$$

Эллипсоид $E_a^n(0) = \{x \in R^n : \frac{x_1^2}{a_1^2} + \dots + \frac{x_n^2}{a_n^2} \leq 1, a_i > 0\}$,

$$c(E_a^n(0), \psi) = \|A\psi\| = \sqrt{a_1^2 \psi_1^2 + \dots + a_n^2 \psi_n^2},$$

где A – диагональная матрица с элементами a_1, \dots, a_n на диагонали.
Эллипсоид $E^n(0) = \{x \in R^n : (Qx, x) \leq 1\}$, где Q – симметричная, положительно определенная матрица порядка n ,

$$c(E^n(0), \psi) = \sqrt{(Q^{-1}\psi, \psi)}.$$

Многогранник $T^m = \text{co}\{x_1, \dots, x_m\}$, $x_i \in R^n$, $i = 1, \dots, m$,

$$c(T^m, \psi) = \max_{i=1, \dots, m} \langle x_i, \psi \rangle,$$

здесь точки x_1, \dots, x_m являются вершинами многогранника.

9 Дополнительные задания

В этом разделе собраны примеры для самостоятельного решения, для которых авторами не было найдено аналитическое решение. Для численного решения необходимо применять компьютерную программу, разработанную в рамках данного практикума. Примеры приведены в том виде, в котором они изначально встречаются в литературе. Соответственно перед началом решения необходимо привести управляемую систему к "каноническому" виду (1), а также выписать матрицу проектирования π , соответствующую каждому терминальному множеству M .

Вместе с каждым примером приводится рисунок зависимости альтернированного интеграла (альтернированных сумм) от времени на заданном отрезке $[0, T]$. Расчет альтернированного интеграла проводился численным методом, описанным в данном методическом пособии, с достаточно высокой точностью.

Пример 1.

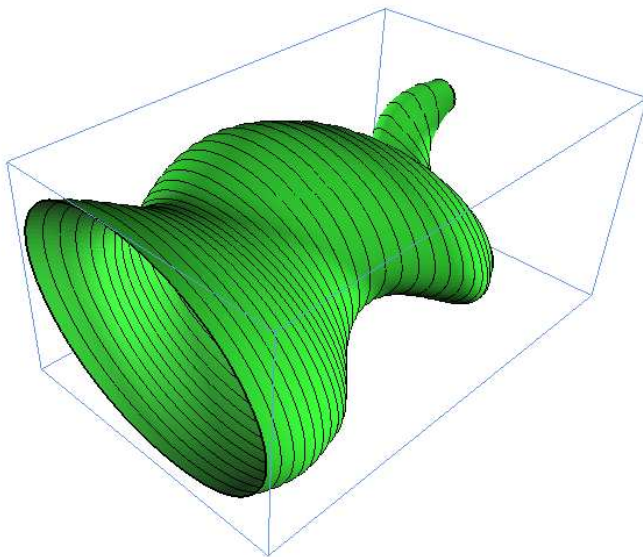
$$\begin{cases} \dot{x}_1 = -0.35x_1 + 2x_2 + v_1, \\ \dot{x}_2 = -x_1 + u_1 + v_2, \end{cases}$$

$$P = \{u_1 : |u_1| \leq 1\},$$

$$Q = \text{co}\{(-1.3, -1.4), (-0.7, 0.6)\},$$

$$M = S_{0.3}(0, 0),$$

$$T = 8.$$



Пример 2.

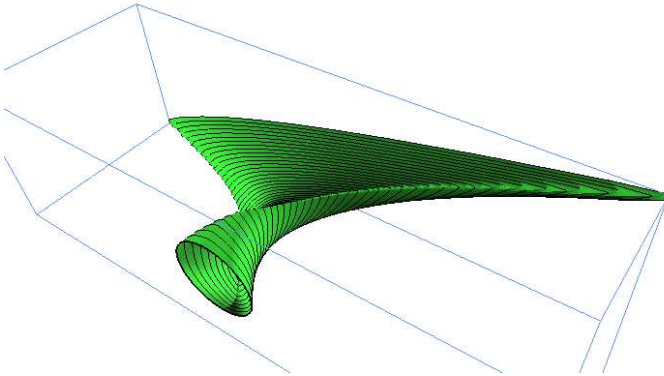
$$\begin{cases} \dot{x}_1 = x_2 + v_1, \\ \dot{x}_2 = u_1, \end{cases}$$

$$P = \{u_1 : |u_1| \leq 1\},$$

$$Q = \{v_1 : |v_1| \leq 0.9\},$$

$$M = S_1(0, 2),$$

$$T = 5.$$



Пример 3.

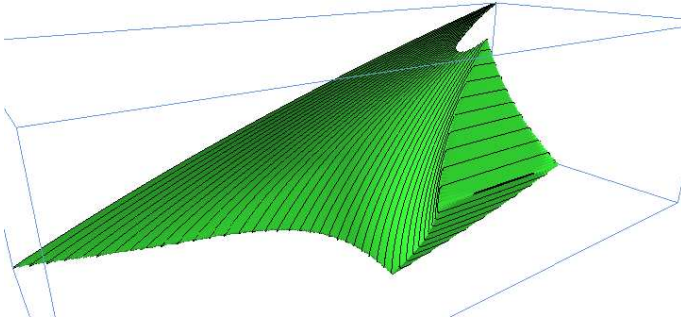
$$\begin{cases} \dot{x}_1 = x_2 + v_1, \\ \dot{x}_2 = u_1, \end{cases}$$

$$P = \{u_1 : |u_1| \leq 1.4\},$$

$$Q = \{v_1 : |v_1| \leq 1\},$$

$$M = \{(x_1, x_2) : |x_1| \leq 1, x_2 = 0\},$$

$$T = 4.$$



Пример 4.

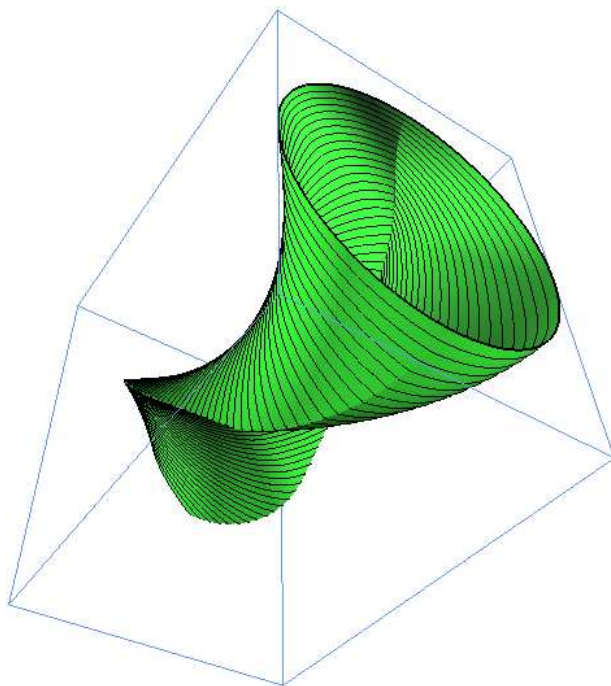
$$\begin{cases} \dot{x}_1 = x_2 + v_1, \\ \dot{x}_2 = -x_1 + u_1, \end{cases}$$

$$P = \{u_1 : |u_1| \leq 1\},$$

$$Q = \{v_1 : |v_1| \leq 0.95\},$$

$$M = S_1(0, 0),$$

$$T = 3.$$



Пример 5.

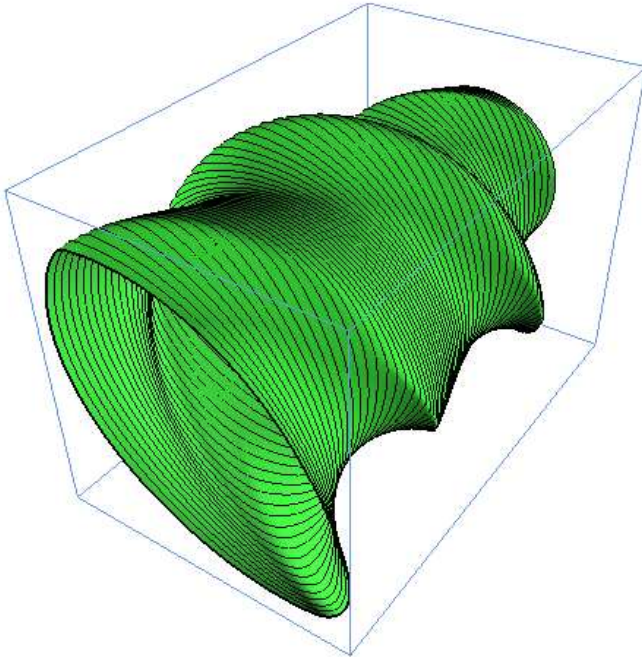
$$\begin{cases} \dot{x}_1 = x_2 + v_1, \\ \dot{x}_2 = -x_1 + u_1 + v_2, \end{cases}$$

$$P = \{u_1 : |u_1| \leq 1.3\},$$

$$Q = \text{co}\{(-0.9, 0), (0, 1), (0.9, 0)\},$$

$$M = S_{1.5}(0, 0),$$

$$T = 8.$$



Пример 6.

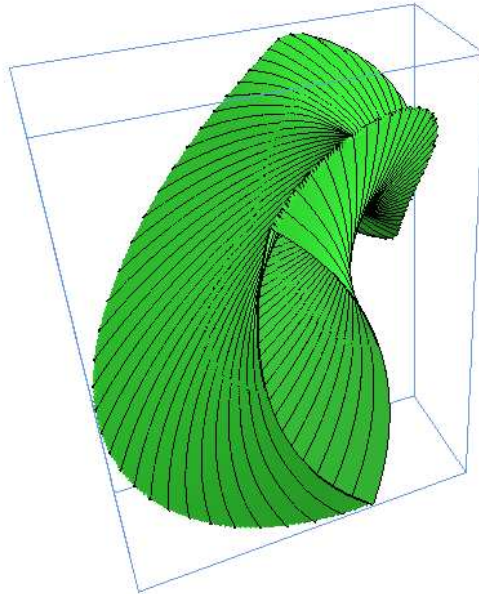
$$\begin{cases} \dot{x}_1 = 0.6x_1 + x_2 + v_1, \\ \dot{x}_2 = -x_1 + u_1, \end{cases}$$

$$P = \{u_1 : -4.6 \leq u_1 \leq 2.5\},$$

$$Q = \{v_1 : -2 \leq v_1 \leq 1.5\},$$

$$M = S_2(0, 0),$$

$$T = 4.$$



Пример 7.

$$\begin{cases} \ddot{x} + 2\dot{x} = u, \\ \ddot{y} + 0.2\dot{y} + y = v, \end{cases}$$

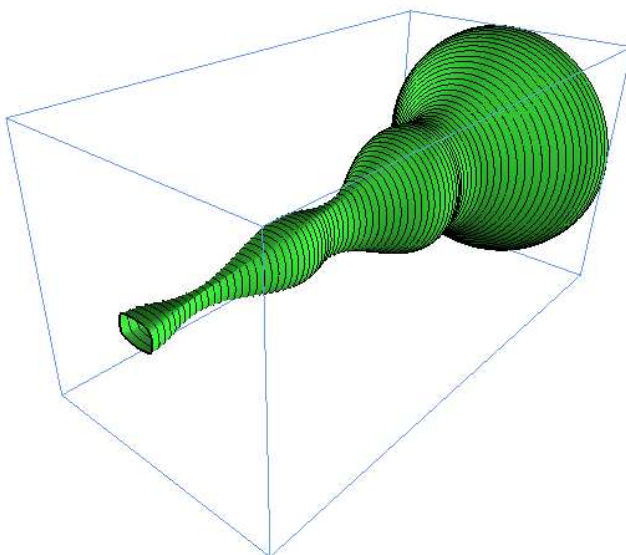
$$x, y \in R^2,$$

$$P = \{u \in R^2 : \frac{u_1^2}{0.8^2} + \frac{u_2^2}{0.4^2} \leq 1\},$$

$$Q = \{v \in R^2 : \frac{v_1^2}{1.5^2} + \frac{v_2^2}{1.05^2} \leq 1\},$$

$$M = \{z \in R^2 : z = x - y, \|z\| \leq 2.5\},$$

$$T = 10.$$



Пример 8.

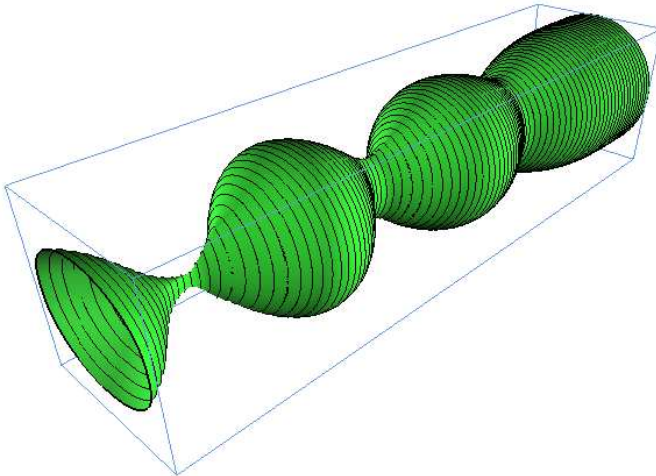
$$\begin{cases} \ddot{x} - 0.025\dot{x} + 1.3x = u, \\ \dot{y} + y = v, \end{cases}$$

$$x, y \in R^2,$$

$$P = Q = \{u \in R^2 : \frac{u_1^2}{1.5^2} + \frac{u_2^2}{1.05^2} \leq 1\},$$

$$M = \{z \in R^2 : z = x - y, \|z\| \leq 1.2\},$$

$$T = 10.$$



Пример 9.

$$\begin{cases} \ddot{x} + 0.025\dot{x} + 1.2x = u, \\ \ddot{y} + 0.01\dot{y} + 0.85y = v, \end{cases}$$

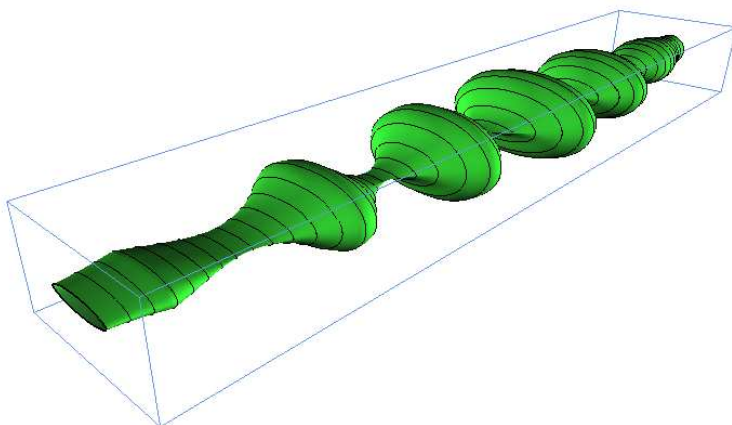
$$x, y \in R^2,$$

$$P = \{u \in R^2 : \frac{u_1^2}{2.0^2} + \frac{u_2^2}{1.3^2} \leq 1\},$$

$$Q = \{v \in R^2 : \frac{v_1^2}{1.5^2} + \frac{v_2^2}{1.05^2} \leq 1\},$$

$$M = \{z \in R^2 : z = x - y, \|z\| \leq 0.4\},$$

$$T = 18.$$



10 Символьные вычисления

Для выполнения символьных вычислений, например, при вычислении значений введенных пользователем опорных функций, предлагается использовать специальную компоненту, разработанную на кафедре Оптимального Управления факультета ВМиК МГУ и предназначенную для использования в среде программирования Delphi.

Компонента представляет собой класс TCalc, располагающийся в файле "Calculator.pas" и обладающий следующими свойствами и методами.

property Variables: TStrings – содержит имена всех, используемых в символьном выражении переменных.

property Formula:String – строка, содержащая символьное выражение, значение которого необходимо вычислить. Перед заданием значения этого свойства необходимо определить все переменные в свойстве *Variables*.

function Calculate(var args: array of Extended): Extended – функция возвращает значение выражения, задаваемого свойством *Formula*. В качестве аргумента функции передается массив значений переменных, используемых в выражении. При этом значения должны идти в том же порядке, в каком сами переменные заданы в свойстве *Variables*.

Символьное выражение должно удовлетворять следующим условиям.

- Заглавные и строчные буквы в выражении не различаются.
- Допустимые операции: +, -, *, /, унарный минус, возведение в степень ^, где степень может быть только числом.
- Допустимые символы в именах переменных: 'A'..'Z', 'a'..'z', '0'..'9'. При этом переменная не может начинаться с цифры.
- Поддерживаемые функции: ln, exp, sqrt, sqr, abs, тригонометрия: cos, sin, tg, ctg, arcsin, arccos, arctg, acctg, ch, sh, th, cth, sgn(sign) – знак числа, h(heavyside) – функция Хэвисайда.

Заметим, что разбор символьного выражения и построение соответствующего ему дерева производится один раз при задании свойства *Formula*. Это позволяет ускорить работу программы при вычислении значения одного и того же выражения при разных значениях переменных.

Ниже приведен пример использования класса TCalc.

```

function UseCalc(X,Y: extended);
var Calc: TCalc;
    Vars: array [1..2] of extended;
begin
    Calc:=TCalc.Create; //Создание переменной типа TCalc
    Calc.Variables.Add('X'); //Добавление новой переменной
    Calc.Variables.Add('Y');
    Calc.Formula:='X+Y'; //Задание символического выражения
    Args[1]:=X; //Значение переменной X
    Args[2]:=Y;
    Result:= Calc.Calculate( Args ); //Вычисление значения выраже-
ния
    Calc.Free; //Освобождение памяти
end;

```

Список литературы

- [1] Понтрягин Л.С. Линейные дифференциальные игры преследования // Матем. сб. 1980. Т. 112. N 3. С. 307–330.
- [2] Понтрягин Л.С., Мищенко А.С. Решение линейной дифференциальной игры преследования на основе альтернированного интегрирования без дискриминации убегающего. ДАН, 1984, т. 277, N 6. С. 1330–1334.
- [3] Никольский М.С. О нижнем альтернированном интеграле Понтрягина в линейных дифференциальных играх преследования // Математический сборник. 1985. Т. 128. N 1. С. 35–49.
- [4] Григоренко Н.Л., Кисилев Ю.Н., Лагунова Н.В., Силин Д.Б., Тринько Н.Г. Методы решения дифференциальных игр двух игроков // Нелинейные динамические системы: качественный анализ и управление. 1994. Вып. 2, С. 49–66.
- [5] Половинкин Е.С., Иванов Г.Е., Балашов М.В., Константинов Р.В., Хорев А.В. Об одном алгоритме численного решения линейных дифференциальных игр // Математический сборник. 2001. Т. 192. N 10. С. 95–122.
- [6] Орлова Г.Б., Силин Д.Б. Приближенное вычисление выпуклой оболочки положительно однородной функции. // Вестник Московского университета. Сер. Вычислительная математика и кибернетика. 1997, № 2, С. 32–55.
- [7] Силин Д.Б., Тринько Н.Г. Модификация алгоритма Грехема для овыпукления положительно-однородной функции // Журнал вычислительной математики и математической физики. 1994. **34**, N 4, С. 631–636.
- [8] Красовский Н.Н., Субботин А.И. Позиционные дифференциальные игры. М.: Наука. 1974.
- [9] Пшеничный Б.Н., Остапенко В.В. Дифференциальные игры. Киев: Наукова думка. 1992.
- [10] Субботин А.И., Ченцов А.Г. Оптимизация гарантии в задачах управления. М.: Наука. 1981.
- [11] Рокафеллер Р. Выпуклый анализ. Мир. 1973.