

Санкт–Петербургский государственный университет

Злобин Данил Юрьевич

Выпускная квалификационная работа

*Квазиоптимальные полиномиальные траектории
в задачах управления подвижных объектов*

Уровень образования: магистратура

Направление 01.04.02 «Прикладная математика и информатика»

Основная образовательная программа ВМ.5517.2019 «Методы
прикладной математики и информатики в задачах управления»

Профиль «Математическое моделирование»

Научный руководитель:


доцент, кафедра механики управляемого движения,
к.ф. - м.н. Шиманчук Дмитрий Викторович

Рецензент:

Военная академия материально-технического обеспечения
имени А. В. Хрулёва,
доцент, кафедра общенаучных дисциплин,
к.т.н. Волков Юрий Александрович

Санкт-Петербург

2021 г.


Злобин Д. Ю.

Содержание

Введение	3
Обзор литературы	4
Постановка задачи	5
Глава 1. Полиномы	7
1.1. Общий вид полинома и мультииндексная нотация	7
1.2. Полиномы и свертки	8
1.3. Оценка области значений полинома на единичном сегменте	10
1.4. Условия включения полинома в выпуклое множество. . .	13
Глава 2. Квази-аппроксимация задач оптимального управле- ния	14
2.1. Постановка задачи	14
2.2. Граничные условия	15
2.3. Сведение к конечномерной задаче нелинейного програм-	
мирования	15
2.4. Градиенты полиномиальных ограничений	17
Глава 3. Пример	21
3.1. Постановка задачи	21
3.2. Вид задачи НЛП	24
3.3. Численный эксперимент	25
Заключение	30
Список литературы	31

Введение

В данной работе исследован вопрос построения допустимой траектории движения нелинейной управляемой механической системы представленной в форме дифференциального включения высшего порядка. Подобная задача возникает в теории управления при построении программного движения, при этом требуется построить программное движение, которое удовлетворяло бы дифференциальное включение (фазовые ограничения) и было бы близко к оптимальному значению некоторого заданного функционала качества. Часто есть смысл рассматривать только полиномиальные траектории системы в пространстве состояний, как, например, в задаче о посадке беспилотного летательного аппарата. Так как полиномиальная траектория бесконечно дифференцируема, то, скорее всего, не будет соответствовать оптимальному управлению, найденному, например, с использованием принципа максимума Понтрягина. Однако, не всегда на практике требуется экстремальное управление, часто требуется лишь приближение к оптимальному в определенном классе функций, и, применительно к данной задаче, метод полиномиальных траекторий может быть успешно использован. Далее будет представлен способ учета фазовых и управляющих ограничений при прямом построении квазиоптимальной полиномиальной траектории в пространстве состояний дифференциального включения высшей степени. Чтобы учесть фазовые ограничения, мы используем асимптотически точную монотонную оценку диапазона значений полинома, основанную на обобщенном разложении по полиномам Бернштейна. Граничные условия учитываются с помощью полинома Эрмита. Представленный метод иллюстрируется на примере задачи терминального управления летательным аппаратом, а именно посадки на движущуюся по заданному закону платформу.

Обзор литературы

Задача управления объектами, поведение которых описывается системами дифференциальных уравнений, находит повсеместное применение на практике. Сложности с получением аналитического решения обуславливают высокую востребованность численных методов. Существует обширные возможности для выбора подходящего численного метода для решения задачи, так, например, для построения численного метода система вида $\dot{x} = f(x, u, t)$ с заданным функционалом качества и ограничениями на управляющее воздействие u обычно сводится к задаче конечномерного нелинейного программирования путем дискретизации, вообще говоря, пространства состояний, управления и множителей Лагранжа соответствующих ограничений [1] (метод коллокаций). Если ограничений нет, то разработаны эффективные методы последовательного приближения, например, [2, 3]. Другим вариантом дискретизации может быть сведение управляемой системы к дифференциальному включению $\dot{x}(t) \in \bigcup_u f(x, u, t)$ и аппроксимации множества достижимости данного включения [4]. Следует заметить, что оба метода успешно используются на практике, например, в NASA. Однако, для метода сведения к дифференциальному включению затруднено получение приближений высшего порядка для множеств достижимости, а метод коллокаций имеет высокую вычислительную сложность и к тому же, как показано в [5], в плохо обусловленных случаях не имеет сильных преимуществ по скорости сходимости. В данной работе предложен метод сведения оптимизационной задачи с ограничением типа дифференциального включения высшего порядка и граничного включения к конечномерной задаче нелинейного программирования, выполнение ограничений которой гарантирует допустимость траектории.

Постановка задачи

В теории управления достаточно общий вид имеет нелинейная система $\dot{\tilde{\chi}}(t) = f(\tilde{\chi}, u, t)$, которую всегда можно представить в стационарной форме $\dot{\chi}(t) = f(\chi, u)$ посредством внесения времени в число переменных пространства состояний. Перед построением численного метода обычно проводят репараметризацию времени, вводят новую переменную нормализованного времени s так, что $t = T s$, где $s \in [0, 1]$, а параметр $T \geq 0$ — финальное время, определяет общее время движения системы, или, иначе говоря, временные рамки в которых рассматривается управляемое движение. Согласно теории дифференциальных включений, а именно, согласно лемме Филиппова, при определенных условиях, данная управляемая система эквивалентна дифференциальному включению

$$\dot{\chi}(t) \in \bigcup_u f(\chi, u),$$

или, учитывая репараметризацию времени,

$$\partial\chi(s) \in \bigcup_u T f(\chi, u),$$

где ∂ есть производная по нормализованному времени s . Используя декартово произведение, данное дифференциальное включение сводится к эквивалентному включению вида

$$(\partial\chi(s), \chi(s)) \in \mathcal{M}_T = \bigcup_{\xi} \bigcup_u (\xi, T f(\xi, u)),$$

Таким образом, предполагая, что χ — вектор столбец, и вводя оператор

$$\mathfrak{D}^n \chi = \begin{bmatrix} \chi \\ \vdots \\ \partial^i \chi \\ \vdots \\ \partial^n \chi \end{bmatrix},$$

можно рассматривать обобщение дифференциальных включений до степени n со свободным параметром T как

$$\mathfrak{D}^n \chi \in \mathcal{M}_T.$$

обобщением граничных условий является граничное включение вида

$$\mathfrak{D}^m \chi(j) \in \mathcal{G}_T^j, \quad j = \overline{0, 1}.$$

Стоит, однако, заметить, что граничное включение не должно вступать в противоречие с дифференциальным включением, иначе оно не имеет смысла. На решениях дифференциального включения степени n , удовлетворяющих граничному включению, таким образом, будем производить минимизацию заданного функционала качества

$$J[\chi(\cdot), T] \longrightarrow \inf.$$

Для того чтобы свести эту задачу к конечномерной форме, мы будем рассматривать только полиномиальные траектории среди тех, которые удовлетворяют названным включениям. Такой подход будем называть методом квазиоптимальных полиномиальных траекторий.

Глава 1. Полиномы

1.1 Общий вид полинома и мультииндексная нотация

При рассмотрении полиномов в данной работе будем подразумевать, что они заданы над конечномерным координатным векторным пространством, и также принимают значения в конечномерном векторном пространстве. Из этого следует, что коэффициенты таких полиномов являются элементами конечномерного пространства области значений. Для записи полиномов будем использовать мультииндексную нотацию. Пусть n — размерность пространства аргумента, тогда рассмотрим $\sigma = (\sigma_1, \dots, \sigma_n)$ — мультииндекс — кортеж натуральных чисел. На универсуме мультииндексов задано поэлементное сложение, вычитание и частичный порядок:

$$\alpha \leq \beta \Leftrightarrow \forall i \alpha_i \leq \beta_i.$$

Удобно использовать следующее правило возведения вектора в мультииндексную степень: $x^\sigma = x_1^{\sigma_1} \cdot \dots \cdot x_n^{\sigma_n}$. Таким образом полагая коэффициенты $b_\sigma \in \mathbb{R}^m$, полиномиальное отображение из пространства \mathbb{R}^n в пространство \mathbb{R}^m , которое мы будем обозначать $p \in \mathcal{P}[\mathbb{R}^n \rightarrow \mathbb{R}^m]$ может быть записано в виде:

$$p(x) = \sum_{\sigma \in \mathbf{S}} b_\sigma x^\sigma,$$

где \mathbf{S} — множество значений мультииндекса. В качестве данного множества можно также брать универсум, полагая часть коэффициентов равными нулю, однако тогда для конечности степени полинома необходимо чтобы множество ненулевых коэффициентов было соответственно конечно, удобнее, однако, полагать это множество изначально заданным и конечным.

Для возведения в мультииндексную степень нам также понадобится

расширение бинорма Ньютона, рассмотрим:

$$\begin{aligned}
(x + y)^\sigma &= (x_1 + y_1)^{\sigma_1} \cdot \dots \cdot (x_n + y_n)^{\sigma_n} \\
&= \sum_{\kappa_1=0}^{\sigma_1} \binom{\sigma_1}{\kappa_1} x_1^{\sigma_1-\kappa_1} y_1^{\kappa_1} \cdot \dots \cdot \sum_{\kappa_n=0}^{\sigma_n} \binom{\sigma_n}{\kappa_n} x_n^{\sigma_n-\kappa_n} y_n^{\kappa_n} \\
&= \sum_{\kappa_1=0}^{\sigma_1} \dots \sum_{\kappa_n=0}^{\sigma_n} \binom{\sigma_1}{\kappa_1} \cdot \dots \cdot \binom{\sigma_n}{\kappa_n} x_1^{\sigma_1-\kappa_1} \cdot \dots \cdot x_n^{\sigma_n-\kappa_n} y_1^{\kappa_1} \cdot \dots \cdot y_n^{\kappa_n} \\
&= \sum_{\kappa \leq \sigma} \binom{\sigma}{\kappa} x^{\sigma-\kappa} y^\kappa,
\end{aligned}$$

где в последней формуле использовано обозначение для мультииндексного биномиального коэффициента, как произведение всех соответствующих биномиальных коэффициентов.

1.2 Полиномы и свертки

Рассмотрим полином одной переменной в одномерное пространство, то есть полином $p \in \mathcal{P}[\mathbb{R} \rightarrow \mathbb{R}]$. Пусть в степенном базисе верно разложение $p(s) = a_0 + \dots + a_d s^d$, тогда определим оператор Υ как координатный оператор над степенным базисом:

$$p \longrightarrow^\Upsilon [a_0 \ \dots \ a_d].$$

Очевидно, что данный оператор является линейным оператором относительно сложения полиномов и умножения на число. Рассматривая произведение полиномов, мы определим свертку векторов как вектор коэффициентов этого произведения. Данное определение соответствует определению свертки последовательностей.

$$\Upsilon [p \cdot q] = \Upsilon [p] * \Upsilon [q] = [a_0 b_0, \ a_0 b_1 + a_1 b_0, \ \dots \ a_d b_r],$$

здесь b_i — коэффициенты полинома q и r — его степень.

Рассмотрим теперь векторный полином $p \in \mathcal{P}[\mathbb{R} \rightarrow \mathbb{R}^m]$. Его коэффициенты векторные, соответственно в качестве значения координатного

оператора будем подразумевать матрицу, в столбцах которой находятся данные коэффициенты, или, иначе, в строках которой расположены коэффициенты соответствующего координатной оси векторного пространства значений полинома. Пусть

$$p(s) = \begin{bmatrix} a_{10} \\ \vdots \\ a_{m0} \end{bmatrix} + \dots + \begin{bmatrix} a_{1d} \\ \vdots \\ a_{md} \end{bmatrix} s^d,$$

тогда

$$\Upsilon[p] = \begin{bmatrix} a_{10} & \dots & a_{1d} \\ \vdots & \ddots & \vdots \\ a_{m0} & \dots & a_{md} \end{bmatrix}.$$

При умножении векторнозначного полинома на скалярнозначный полином размерность пространства значений сохраняется. Для определения векторных коэффициентов данного произведения мы определим свертку матрицы с вектор-строкой как матрицу, в которой каждая строка является сверткой соответствующей строки исходной матрицы с исходной вектор-строкой. Иначе говоря, если r — скалярнозначный полином и

$$\Upsilon[r] = \begin{bmatrix} r_0 & \dots & r_{d_r} \end{bmatrix},$$

то

$$\begin{bmatrix} a_{10} & \dots & a_{1d} \\ \vdots & \ddots & \vdots \\ a_{m0} & \dots & a_{md} \end{bmatrix} * \begin{bmatrix} r_0 & \dots & r_{d_r} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} a_{10} & \dots & a_{1d} \end{bmatrix} * \begin{bmatrix} r_0 & \dots & r_{d_r} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} a_{m0} & \dots & a_{md} \end{bmatrix} * \begin{bmatrix} r_0 & \dots & r_{d_r} \end{bmatrix} \end{bmatrix},$$

тогда и в этом случае выражение $\Upsilon[p \cdot r] = \Upsilon[p] * \Upsilon[r]$ будет верным.

Для заданной матрицы определим мультисверточную степень по аналогии с мультииндексной степенью вектора. Рассмотрим мультииндекс $\sigma =$

$(\sigma_1, \dots, \sigma_m)$, тогда

$$\begin{bmatrix} a_{10} & \dots & a_{1d} \\ \vdots & \ddots & \vdots \\ a_{m0} & \dots & a_{md} \end{bmatrix}^{*\sigma} = \begin{bmatrix} a_{10} & \dots & a_{1d} \end{bmatrix}^{*\sigma_1} * \dots * \begin{bmatrix} a_{m0} & \dots & a_{md} \end{bmatrix}^{*\sigma_m}.$$

Данный подход потребуется для нахождения вектора коэффициентов композиций полиномов в дальнейшем. Как можно убедиться, коэффициенты композиции полинома $q \in \mathcal{P}[\mathbb{R}^m \rightarrow \mathbb{R}^l]$ и полинома p вычисляются как

$$\Upsilon[q(p(s))] = \sum_{\sigma \in \mathbf{S}_q} b_\sigma * \Upsilon[p]^{*\sigma},$$

если верно представление

$$q(x) = \sum_{\sigma \in \mathbf{S}_q} b_\sigma x^\sigma,$$

где b_σ — вектор-столбцы коэффициентов, которые можно воспринимать как полиномы нулевой степени. Операции свертки с вектор-строкой, в таком случае, соответствует произведению Кронекера вектор-столбца на вектор-строку справа. Также условимся, что при сложении векторов или матриц разной размерности вдоль строк, они дополняются нулями справа, что непосредственно соответствует сложению полиномов.

1.3 Оценка области значений полинома на единичном сегменте

Рассмотрим произвольный полином $q \in \mathcal{P}[\mathbb{R} \rightarrow \mathbb{R}]$. Как показано в работах [8, 9], область значений полинома на единичном сегменте $s \in [0, 1]$ может быть оценена с помощью разложения q по базису полиномов Бернштейна. Для того чтобы сделать это, мы выбираем произвольное целое число $k \geq d$, где $d = \deg q$, и записываем полином в форме Бернштейна:

$$q(s) = \sum_{i=0}^k b_i^k \binom{k}{i} s^i (1-s)^{k-i}. \quad (1)$$

Коэффициенты этой формы вычисляются с помощью степенного разложения полинома q следующим образом [8]:

$$b_i^k = \sum_{l=0}^{\min(i,d)} a_l \binom{i}{l} / \binom{k}{l}. \quad (2)$$

Теперь, используя форму Бернштейна, (1), мы можем записать оценку области значений полинома, следует учесть что точность оценки зависит от параметра k .

$$\min_{0 \leq i \leq k} b_i^k \leq q(s) \leq \max_{0 \leq i \leq k} b_i^k \underbrace{\sum_{i=0}^k \binom{k}{i} s^i (1-s)^{k-i}}_{(s+1-s)^k=1} = \max_{0 \leq i \leq k} b_i^k. \quad (3)$$

Самая главная особенность указанной оценки (3), состоит в том, что благодаря параметру k , как показано в работе [9], она асимптотически точна в указанном далее смысле:

$$\begin{aligned} \lim_{k \rightarrow \infty} \min_{0 \leq i \leq k} b_i^k &= \min_{s \in [0,1]} q(s), \\ \lim_{k \rightarrow \infty} \max_{0 \leq i \leq k} b_i^k &= \max_{s \in [0,1]} q(s). \end{aligned}$$

Мы используем данные оценки для построения асимптотически точных ограничений на коэффициенты полинома q таким образом, что значения полинома лежат в заданной допустимой области когда аргумент полинома пробегает значения из сегмента $s \in [0, 1]$. Допустим, мы хотим обеспечить условие $q^- \leq q(s) \leq q^+$, данное ограничение будет гарантированно выполнено, если

$$q^- \leq \min_{0 \leq i \leq k} b_i^k \leq \max_{0 \leq i \leq k} b_i^k \leq q^+,$$

что эквивалентно

$$\forall i = \overline{0, k} \quad q^- \leq b_i^k \leq q^+. \quad (4)$$

Обозначим коэффициенты

$$\lambda_{li}^k = \binom{i}{l} / \binom{k}{l} = \frac{i!}{l!(i-l)!} \cdot \frac{(k-l)!l!}{k!} = \frac{(i-l+1) \cdot \dots \cdot i}{(k-l+1) \cdot \dots \cdot k} = \frac{(i)_l}{(k)_l}.$$

Теперь мы можем записать (4) в матричной форме, используя выражение (2):

$$\begin{bmatrix} b_0^k & b_1^k & \dots & b_d^k & \dots & b_k^k \end{bmatrix} = \begin{bmatrix} a_0 & \dots & a_d \end{bmatrix} \underbrace{\begin{bmatrix} \lambda_{00}^k & \lambda_{01}^k & \dots & \lambda_{0d}^k & \dots & \lambda_{0k}^k \\ 0 & \lambda_{11}^k & \dots & \lambda_{1d}^k & \dots & \lambda_{1k}^k \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \lambda_{dd}^k & \dots & \lambda_{dk}^k \end{bmatrix}}_{\Lambda^{d,k}},$$

и соответственно ограничения принимают следующий вид:

$$q^- \leq \Upsilon [q] \Lambda^{d,k} \leq q^+. \quad (5)$$

Сравнение вектора и числа подразумевается в покоординатном смысле. Следовательно, если полином удовлетворяет выражению (5), тогда область его значений включена в сегмент $[q^-, q^+]$. С другой стороны, при $k \rightarrow \infty$ неравенство (5) включает в себя все полиномы, чья область значений включена в сегмент $[q^-, q^+]$, что гарантируется асимптотической точностью оценок (следует понимать, что для некоторых полиномов, возможно не существует конечного k , такого, чтобы они удовлетворяли указанному условию, такие полиномы, например, существуют среди тех, которые достигают значений q^-, q^+ при некотором s). Следовательно мы получаем необходимые (в пределе) и достаточные (всегда) условия для того, чтобы при $s \in [0, 1]$ интервал значений полинома был включен в $[q^-, q^+]$.

Все сказанное выше напрямую обобщается на векторнозначные полиномы $q \in \mathcal{P} [\mathbb{R} \rightarrow \mathbb{R}^l]$, для получения условий включения векторнозначного полинома в область, разложимую на декартово произведение одномерных сегментов. Для этого в выражении (5) необходимо использовать определение координатного оператора как указывалось ранее — матрицы

в столбцах которой находятся векторные коэффициенты полинома. Далее для простоты будем использовать отображение $\Lambda^{d,k}$ без указания степени d .

1.4 Условия включения полинома в выпуклое множество.

Рассмотрим выпуклое множество \mathcal{M}^c в пространстве значений векторнозначного полинома $q \in \mathcal{P} [\mathbb{R} \rightarrow \mathbb{R}^l]$. Так как данное множество выпуклое, то $q([0, 1]) \subset \mathcal{M}^c$ эквивалентно $\text{conv } q([0, 1]) \subset \mathcal{M}^c$, где conv означает выпуклую оболочку. Совершенно аналогично комплексному случаю, представленному в статье [9], для $q \in \mathcal{P} [\mathbb{R} \rightarrow \mathbb{R}^l]$ можно показать, что если b_0^k, \dots, b_k^k коэффициенты формы Бернштейна заданного полинома, тогда

$$\text{conv } q([0, 1]) = \bigcap_{j=d}^{\infty} \text{conv}\{b_0^j, \dots, b_j^j\} \subset \text{conv}\{b_0^k, \dots, b_k^k\}.$$

Тогда, задавая $\text{conv}\{b_0^k, \dots, b_k^k\} \subset \mathcal{M}^c$, мы получаем асимптотически точные условия на включение полинома в выпуклое множество на единичном сегменте. Эквивалентным образом эти условия выражаются как

$$\forall i = \overline{0, k} \quad b_i^k \in \mathcal{M}^c.$$

И если множество \mathcal{M}^c может быть представлено как

$$\mathcal{M}^c = \{x \in \mathbb{R}^l \mid f(x) \leq 0, g(x) = 0\}, \quad (6)$$

тогда условия выражаются в следующем виде

$$\forall i = \overline{0, k} \quad f(b_i^k) \leq 0, \quad g(b_i^k) = 0, \quad (7)$$

где функции, задающие множество, могут быть неполиномиальны.

Глава 2. Квази-аппроксимация задач оптимального управления

2.1 Постановка задачи

Рассмотрим стационарное дифференциальное включение заданной степени с параметром T , который соответствует финальному времени.

$$\mathfrak{D}^n \chi \in \mathcal{M}_T, \quad (8)$$

подразумевается, что траектория $\chi(s)$ задана нормализованным временем $s \in [0, 1]$ в пространстве \mathbb{R}^l , и параметр T соответствует общему времени движения вдоль траектории. На траекторию наложено граничное включение, которое совместно с дифференциальным включением

$$\mathfrak{D}^m \chi(j) \in \mathcal{G}_T^j, \quad j = \overline{0, 1}. \quad (9)$$

Конкретный вид множества \mathcal{M}_T может быть очень сложен, в нашем исследовании мы ограничимся предположением, что $\mathcal{M}_T = \mathcal{M}_T^p \cap \mathcal{M}_T^c$. Где \mathcal{M}_T^c — это произвольное выпуклое множество, и \mathcal{M}_T^p — множество с алгебраической границей, в следующем смысле

$$\mathcal{M}_T^p = \left\{ \xi \in \mathbb{R}^{l(n+1)} \mid q(\xi | T) \leq 0, w(\xi | T) = 0 \right\}. \quad (10)$$

Где q и w это полиномы $q(\cdot | T) \in \mathcal{P} [\mathbb{R}^{l(n+1)} \rightarrow \mathbb{R}^{l_1}]$, $w(\cdot | T) \in \mathcal{P} [\mathbb{R}^{l(n+1)} \rightarrow \mathbb{R}^{l_2}]$, коэффициенты которых параметризованы финальным временем. Пусть также множество \mathcal{M}_T^c может быть представлено как

$$\mathcal{M}_T^c = \left\{ \xi \in \mathbb{R}^{l(n+1)} \mid f(\xi | T) \leq 0, g(\xi | T) = 0 \right\}. \quad (11)$$

Учитывая все сказанное ранее, мы рассмотрим проблему минимизации заданного функционала J

$$J[\chi(\cdot), T] \longrightarrow \inf. \quad (12)$$

Мы ограничимся рассмотрением только полиномиальных траекторий. Для этого в число ограничений формально добавим следующее выражение

$$\chi(\cdot) \in \mathcal{P} [\mathbb{R} \rightarrow \mathbb{R}^l]. \quad (13)$$

Проблему (8)–(12) вместе с условием (13) назовем *квазиаппроксимацией* исходной проблемы.

2.2 Граничные условия

Ограничения (9) граничные для полинома (13). Из теории кратной интерполяции известно, что полином минимальной степени, удовлетворяющий данным условиям — полином Эрмита [6, 7], который в нашем случае может быть записан следующим образом

$$\begin{aligned} \chi_H(s | \mathfrak{D}^m \chi(0|1)) &= \sum_{j=0}^1 \sum_{i=0}^m \partial^i \chi(j) \mu_{ij}(s), \\ \mu_{ij}(s) &= \sum_{q=0}^{m-i} \frac{(-1)^{(q+n+1)j}}{i!} \binom{n+q}{q} (s-j)^{i+q} (1-j-s)^{n+1}. \end{aligned} \quad (14)$$

Подставляя граничные условия в выражение (14), мы получаем

$$\chi_H(s | v) = \sum_{j=0}^1 \sum_{i=0}^m v^{ij} \mu_{ij}(s), \quad v^j = (v^{0j}, \dots, v^{mj}) \in \mathcal{G}_T^j$$

Теперь любой полином (13), удовлетворяющий граничным условиям (9) может быть единственным образом выражен через полином Эрмита:

$$\chi(s | v) = \chi_H(s | v) + s^{m+1} (s-1)^{m+1} r(s), \quad r(\cdot) \in \mathcal{P} [\mathbb{R} \rightarrow \mathbb{R}^l]. \quad (15)$$

2.3 Сведение к конечномерной задаче нелинейного программирования

Мы начинаем с выражений (15), (5) и (7). Предположим, что хотим найти квазиоптимальную траекторию степени $2m+1+d$, тогда рассмотрим

векторные коэффициенты $a = (a^0, \dots, a^{d-1})$, $a^i \in \mathbb{R}^l$. В соответствии с (15)

$$\chi(s | v, a) = \chi_H(s | v) + s^{m+1}(s-1)^{m+1}(a^0 + \dots + a^{d-1}s^{d-1}), \quad v^j \in \mathcal{G}_T^j. \quad (16)$$

Выражение (16) представляет собой параметризацию всех полиномов степени не выше $2m + 1 + d$ среди тех, которые удовлетворяют граничным условиям. Мы используем эту параметризацию чтобы свести исходную задачу к конечномерной форме. Предположим, что в дополнение к (10) и (11) следующее представление имеет место быть

$$\mathcal{G}_T^j = \left\{ \xi \in \mathbb{R}^{l(m+1)} \mid h_j(\xi | T) \leq 0, r_j(\xi | T) = 0 \right\}.$$

Тогда исходная задача, в соответствии с изложенным выше, сводится к форме

$$\inf_{a, v, T} J[\chi(\cdot | v, a), T], \quad (17)$$

$$\Upsilon[q(\mathfrak{D}^n \chi(s | v, a) | T)] \Lambda^k \leq 0, \quad \Upsilon[w(\mathfrak{D}^n \chi(s | v, a) | T)] = 0, \quad (18)$$

$$\begin{aligned} f(\Upsilon[\mathfrak{D}^n \chi(s | v, a)] \lambda_i^k | T) \leq 0, & \quad g(\Upsilon[\mathfrak{D}^n \chi(s | v, a)] \lambda_i^k | T) = 0, \\ h_j(v^j | T) \leq 0, & \quad r_j(v^j | T) = 0, \end{aligned} \quad (19)$$

$$\begin{aligned} k \text{ is fixed and big enough, } & \quad i = \overline{0, k}, \\ \lambda_i^k \text{ is the } i\text{-th column of } \Lambda^k, & \quad j = \overline{0, 1}, \\ a \in \mathbb{R}^{ld}, v = (v^0, v^1) \in \mathbb{R}^{2(m+1)l}, & \quad T \geq 0. \end{aligned}$$

Как можно заметить, проблема (17)–(19) — это задача нелинейного программирования. Число переменных по которым ведется минимизация — $ld + 2(m+1)l + 1$. Как показывает опыт использования данного метода, для решения такой задачи разумно использовать метод *SQP* [10], который сходится значительно быстрее нежели, к примеру, метод *interior-point* [11]. Так же уточним, что применительно к (18), можно проводить пре-

образования и представлять левые части выражений как многочлены от коэффициентов исходных многочленов, а умножение будет определяться как операция свертки над последовательностью коэффициентов исходных многочленов. Рассмотрим подробнее структуру данных ограничений.

2.4 Градиенты полиномиальных ограничений

Рассмотрим для примера ограничения типа неравенства из группы (18). Так как q — полином, то он допускает представление с использованием мультииндексной нотации

$$q(x | T) = \sum_{\sigma \in \mathbf{S}_q} b_\sigma(T) x^\sigma.$$

Положим $x = x(s | a, v) = \mathfrak{D}^n \chi(s | v, a)$, данное выражение конструктивно, оно может быть вычислено на основании предыдущих результатов посредством операции дифференцирования и использования формулы (16). Так как оператор \mathfrak{D}^n линеен, то x представим как сумма $x = x_h + x_p$, где первое слагаемое соответствует полиному Эрмита и зависит только от граничных условий, а второе слагаемое зависит только от векторных коэффициентов a^i . Перейдем от полиномов к матрицам их коэффициентов, тогда предыдущее выражение примет вид

$$\gamma_q = \sum_{\sigma \in \mathbf{S}_q} b_\sigma * \gamma_x^{*\sigma},$$

где $\gamma_q = \Upsilon[q(x(s | a, v))]$, $\gamma_x = \Upsilon[x(s | a, v)]$. Звездочкой, как и ранее, обозначена операция свертки. Примем соглашение до конца данного пункта — опускать звездочку для простоты записи. Далее неуказанные операции умножения считаются операцией свертки, а также опустим явное указание зависимостей. Используя обобщенный бином Ньютона, запишем ограничение в следующем виде

$$\gamma_q = \sum_{\sigma \in \mathbf{S}_q} b_\sigma (\gamma_h + \gamma_p)^\sigma = \sum_{\sigma \in \mathbf{S}_q} \sum_{\kappa \leq \sigma} \binom{\sigma}{\kappa} b_\sigma \gamma_h^{\sigma - \kappa} \gamma_p^\kappa.$$

Операции суммирования в данном выражении можно поменять местами, построив соответствующие множества $\mathbf{M}_q = \cup_{\sigma \in \mathbf{S}_q} \{\kappa \leq \sigma\}$ и для каждого κ множество \mathbf{S}_κ , так что

$$\gamma_q = \sum_{\kappa \in \mathbf{M}_q} \underbrace{\left(\sum_{\sigma \in \mathbf{S}_\kappa} \binom{\sigma}{\kappa} b_\sigma \gamma_h^{\sigma-\kappa} \right)}_{A_\kappa} \gamma_p^\kappa.$$

Видно, что ограничения имеют сложную полиномиальную структуру. Для успешного решения задачи с такими ограничениями в общем случае, оптимизационный алгоритм должен эффективно работать с невыпуклыми функциями. Также данная форма представления ограничений удобна тем, что мы провели разделение: выражение в скобках (матрица A_κ) зависит только от финального времени и граничных условий, а выражение снаружи только от векторных коэффициентов a^i . При таком представлении не требуется пересчитывать выражение в скобках при поиске производных. Это позволяет экономить количество операций при вычислении производных, так как все выражения в скобках должны быть вычислены на этапе вычисления самого γ_q и могут быть использованы повторно. Найдем вид производных для использования градиентных методов оптимизации.

$$\begin{aligned} \frac{\partial \gamma_q}{\partial a_j^i} &= \sum_{\kappa \in \mathbf{M}_q} A_\kappa \frac{\partial \gamma_p^\kappa}{\partial a_j^i}, \\ \frac{\partial \gamma_p^\kappa}{\partial a_j^i} &= \sum_{r=1}^{l(n+1)} \kappa_r \gamma_p^{\kappa-\mathbf{1}_r} \frac{\partial \gamma_{p,r}}{\partial a_j^i}, \end{aligned}$$

здесь $\gamma_{p,r}$ — строка r матрицы γ_p , $\mathbf{1}_r$ — мультииндекс со всеми нулями, кроме единицы в r -той позиции. Также обозначим многочлен $\omega = s^{m+1}(s-1)^{m+1}$, который обычно называют узловым для задачи построения полинома Эрмита. Продолжим вычисление производных, необходимых для преды-

дущего выражения

$$\begin{aligned}
\frac{\partial \gamma_{p,r}}{\partial a_j^i} &= \frac{\partial}{\partial a_j^i} \Upsilon [\partial^{r_1} (\omega (a_{r_2}^0 + \dots + a_{r_2}^i s^i + \dots a_{r_2}^d s^d))] \\
&= \Upsilon \left[\partial^{r_1} \left(\omega \frac{\partial}{\partial a_j^i} (a_{r_2}^0 + \dots + a_{r_2}^i s^i + \dots a_{r_2}^d s^d) \right) \right] \\
&= \delta_{jr_2} \underbrace{\Upsilon [\partial^{r_1} (\omega s^i)]}_{d_{r_1 i}},
\end{aligned} \tag{20}$$

при этом $r = r_1 l + r_2$ такое разложение использовано из-за блочной структуры при действии оператора \mathfrak{D}^n . Конкретнее

$$\mathfrak{D}^n \chi = \begin{bmatrix} \chi_1 \\ \vdots \\ \chi_l \\ \vdots \\ \partial^n \chi_1 \\ \vdots \\ \partial^n \chi_l \end{bmatrix}.$$

Среди производных (20) не равны нулю только производные при $r \in \mathbf{I}_j = \{r = j + lz \mid z = \overline{0, n}\}$, таким образом, окончательно получаем

$$\frac{\partial \gamma_q}{\partial a_j^i} = \sum_{z=0}^n \sum_{\kappa \in \mathbf{M}_q} A_\kappa \kappa_{j+lz} \gamma_p^{\kappa - \mathbf{1}_{j+lz}} d_{zi},$$

совершенно аналогично можно найти

$$\begin{aligned}
\frac{\partial \gamma_q}{\partial v_j^{ir}} &= \sum_{z=0}^n \sum_{\kappa \in \mathbf{M}_q} B_\kappa \kappa_{j+lz} \gamma_h^{\kappa - \mathbf{1}_{j+lz}} c_{zir}, \\
B_\kappa &= \sum_{\sigma \in \mathbf{S}_\kappa} \binom{\sigma}{\kappa} b_\sigma \gamma_p^{\sigma - \kappa}, \\
c_{zir} &= \Upsilon [\partial^z \mu_{ir}].
\end{aligned}$$

Напомним, что μ_{ir} — многочлены из выражения (14) для полинома Эрмита. Таким образом осталось лишь домножить матрично получившийся результат на Λ^k справа и будут найдены градиенты полиномиальных ограничений, для остальных ограничений нахождение градиентов не составляет особого труда, поэтому не приводится в работе.

Глава 3. Пример

3.1 Постановка задачи

Приведем пример применения изложенного метода для построения квазиоптимальной траектории посадки летательного аппарата с целью минимизации времени посадки. Ранее данная проблема была рассмотрена в [13], где была построена Эрмитова траектория, однако, фазовые ограничения проверялись в конечном числе точек, а между ними допустимость не гарантировалась.

$$\begin{cases} \dot{h} = v \sin(\alpha), \\ \dot{v} = g (n_1 - \sin(\alpha)), \\ \dot{y} = v \cos(\alpha) \cos(\beta), \\ \dot{\alpha} = g \frac{n_2 \cos(\gamma) - \cos(\alpha)}{v}, \\ \dot{z} = -v \cos(\alpha) \sin(\beta), \\ \dot{\beta} = -g \frac{n_2 \sin(\gamma)}{v \cos(\alpha)}. \end{cases} \quad (21)$$

Здесь h — высота полета, y — дальность, z — поперечное смещение, v — скорость, α — угол наклона траектории, β — путевой угол, γ — угол крена, n_1 — продольная перегрузка, n_2 — поперечная перегрузка, g — ускорение свободного падения. В данной модели в качестве управляющего воздействия рассматривается вектор $[n_1, n_2, \gamma]^\top$. Заменой переменных $t = Ts$, $s \in [0, 1]$ и

$$\begin{aligned} u_1 &= n_2 \sin(\gamma), & \xi_1 &= v \sin(\alpha), & \chi_1 &= h, \\ u_2 &= n_2 \cos(\gamma), & \xi_2 &= v \cos(\alpha) \cos(\beta), & \chi_2 &= y, \\ u_3 &= n_1, & \xi_3 &= v \cos(\alpha) \sin(\beta), & \chi_3 &= z, \end{aligned}$$

система (21) сводится к

$$\frac{1}{g} \left(\frac{\partial^2 \chi}{T^2} - f_c \right) = B(\partial \chi) u, \quad \partial \chi_2^2 + \partial \chi_3^2 \neq 0, \quad (22)$$

с матрицей

$$B(\partial\chi) = \begin{pmatrix} 0 & \frac{\sqrt{\partial\chi_2^2 + \partial\chi_3^2}}{\|\partial\chi\|} & \frac{\partial\chi_1}{\|\partial\chi\|} \\ -\partial\chi_3 & -\frac{\partial\chi_1\partial\chi_2}{\sqrt{\partial\chi_2^2 + \partial\chi_3^2}} & \frac{\partial\chi_2}{\|\partial\chi\|} \\ \frac{\partial\chi_2}{\sqrt{\partial\chi_2^2 + \partial\chi_3^2}} & \frac{-\partial\chi_1\partial\chi_3}{\sqrt{\partial\chi_2^2 + \partial\chi_3^2}} & \frac{\partial\chi_3}{\|\partial\chi\|} \\ \frac{\partial\chi_2}{\sqrt{\partial\chi_2^2 + \partial\chi_3^2}} & \frac{-\partial\chi_1\partial\chi_3}{\sqrt{\partial\chi_2^2 + \partial\chi_3^2}} & \frac{\partial\chi_3}{\|\partial\chi\|} \end{pmatrix}.$$

Кроме того $\partial = \frac{d}{ds}$, $f_c = [-g, 0, 0]^\top$. Легко убедиться, что B является матрицей вращения, следовательно, B не изменяет норму вектора управляющего воздействия u . Пусть на движение БПЛА накладываются ограничения следующего вида, во-первых это ограничения в пространстве состояний:

$$\begin{cases} C_v^- \leq \|\dot{\chi}\| \leq C_v^+, \\ C_u^- \leq \|u\| \leq C_u^+, \\ C_h^- \leq \chi_1 \leq C_h^+, \end{cases}$$

на скорость полета, на управляющее воздействие и на высоту полета соответственно. Во-вторых граничные условия, которые состоят из начальных и конечных условий, причем конечные условия могут зависеть от финального времени

$$\begin{cases} \chi(0) = \Gamma^{0,0}, & \chi(T) = \Gamma^{0,1}(T), \\ \dot{\chi}(0) = \Gamma^{1,0}, & \dot{\chi}(T) = \Gamma^{1,1}(T), \\ \ddot{\chi}(0) = \Gamma^{2,0}, & \ddot{\chi}(T) = \Gamma^{2,1}(T). \end{cases}$$

Переходя к дифференцированию по нормализованному времени s а также подставляя управление из выражения (22) и возводя в квадрат норму, получаем ограничения в пространстве состояний в следующем виде

$$\begin{cases} C_v^{-,2} \leq \frac{\partial\chi^2}{T^2} \leq C_v^{+,2}, \\ C_u^{-,2} \leq \frac{1}{g^2} \left(\frac{\partial^2\chi}{T^2} - f_c \right)^2 \leq C_u^{+,2}, \\ C_h^- \leq \chi_1 \leq C_h^+, \end{cases}$$

данные ограничения являются полиномиальными и определяют множество с алгебраической границей в девятимерном пространстве, выпишем в явном виде полиномы, определяющие это множество

$$\begin{aligned}
q_1(x | T) &= \frac{1}{T^2}x_4^2 + \frac{1}{T^2}x_5^2 + \frac{1}{T^2}x_6^2 - C_v^{+,2}, \\
q_1(x | T) &= C_v^{-,2} - \frac{1}{T^2}x_4^2 - \frac{1}{T^2}x_5^2 - \frac{1}{T^2}x_6^2, \\
q_3(x | T) &= \frac{1}{g^2T^4}x_7^2 + \frac{1}{g^2T^4}x_8^2 + \frac{1}{g^2T^4}x_9^2 + \frac{2}{gT^2}x_7 + 1 + C_u^{+,2}, \\
q_4(x | T) &= C_u^{+,2} - \frac{1}{g^2T^4}x_7^2 - \frac{1}{g^2T^4}x_8^2 - \frac{1}{g^2T^4}x_9^2 - \frac{2}{gT^2}x_7 - 1, \\
q_5(x | T) &= x_1 - C_h^+, \\
q_6(x | T) &= C_h^- - x_1.
\end{aligned} \tag{23}$$

Тогда ограничения в пространстве состояний вместе с управляемой системой (22) являются непосредственно дифференциальным включением вида

$$\mathfrak{D}^2\chi \in \mathcal{M}_T^p = \{x | q(x | T) \leq 0\}.$$

Граничные условия после преобразования времени к нормализованному и записи в виде граничного включения с одноточечным множеством принимают форму

$$\mathfrak{D}^2\chi(j) \in \mathcal{G}_T^j = \{(\Gamma^{0,j}, T\Gamma^{1,j}, T^2\Gamma^{2,j})\}, \quad j = \overline{0,1}.$$

Таким образом, определяя функционал качества как финальное время

$$J[\chi(\cdot), T] = T \longrightarrow \inf,$$

приходим непосредственно к задаче вида, рассмотренного в предыдущей главе работы, и, соответственно, представленный метод может быть использован для поиска квазиоптимальной траектории.

3.2 Вид задачи НЛП

Непосредственно для данной задачи можно вычислить базисные полиномы μ_{ij} , фигурирующие в выражении (14) для полинома Эрмита,

$$\begin{aligned}\mu_{00} &= -6s^5 + 15s^4 - 10s^3 + 1, \\ \mu_{10} &= -3s^5 + 8s^4 - 6s^3 + s, \\ \mu_{20} &= -\frac{1}{2}s^5 + \frac{3}{2}s^4 - \frac{3}{2}s^3 + \frac{1}{2}s^2, \\ \mu_{01} &= 6s^5 - 15s^4 + 10s^3, \\ \mu_{11} &= -3s^5 + 7s^4 - 4s^3, \\ \mu_{21} &= \frac{1}{2}s^5 - s^4 + \frac{1}{2}s^3.\end{aligned}$$

Тогда полином Эрмита в явном виде представляется как

$$\begin{aligned}\chi_H(s|T) &= \left[6(\Gamma^{0,1} - \Gamma^{0,0}) - 3T(\Gamma^{1,0} + \Gamma^{1,1}) + \frac{1}{2}T^2(\Gamma^{2,1} - \Gamma^{2,0}) \right] s^5 + \\ &\quad \left[15(\Gamma^{0,0} - \Gamma^{0,1}) + T(8\Gamma^{1,0} + 7\Gamma^{1,1}) + T^2\left(\frac{3}{2}\Gamma^{2,0} - \Gamma^{2,1}\right) \right] s^4 + \\ &\quad \left[10(\Gamma^{0,1} - \Gamma^{0,0}) - T(6\Gamma^{1,0} + 4\Gamma^{1,1}) + \frac{1}{2}T^2(\Gamma^{2,1} - 3\Gamma^{2,0}) \right] s^3 + \\ &\quad \frac{1}{2}T^2\Gamma^{2,0}s^2 + T\Gamma^{1,0}s + \Gamma^{0,0}.\end{aligned}$$

Обозначим выражения в скобках перед степенью s^i как h_i , напомним, что это трехмерный вектор столбец который зависит от финального времени. Тогда матрица $\gamma_h = \Upsilon [\mathfrak{D}^2\chi_H]$ имеет следующий вид

$$\gamma_h = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & h_4 & h_5 \\ h_1 & 2h_2 & 3h_3 & 4h_4 & 5h_5 & 0 \\ 2h_2 & 6h_3 & 12h_4 & 20h_5 & 0 & 0 \end{bmatrix}_{9 \times 6},$$

а матрица $\gamma_p = \Upsilon [\mathfrak{D}^2(s^3(s-1)^3a^0)]$, которая зависит, к примеру, от единственного векторного коэффициента a^0 ,

$$\gamma_p = \begin{bmatrix} 0 & 0 & 0 & a^0 & -3a^0 & 3a^0 & -a^0 \\ 0 & 0 & 3a^0 & -12a^0 & 15a^0 & -6a^0 & 0 \\ 0 & 6a^0 & -36a^0 & 60a^0 & -30a^0 & 0 & 0 \end{bmatrix}_{9 \times 7}.$$

Напомним, что при сложении данные матрицы дополняются справа нулями, что находится в соответствии со сложением полиномов. В общем случае, при большем количестве коэффициентов a^i , матрица γ_p выглядит намного сложнее и явное выражение для нее не приводится в виду громозкости, тем не менее, однако, данная матрица может быть легко вычислена. Допустим что мы переписали (23) в виде $q = \sum_{\sigma \in \mathbf{S}_q} b_\sigma(T) x^\sigma$, тогда в соответствии с методом задача НЛП с одним векторны коэффициентом имеет вид

$$T \longrightarrow \inf, \left[\sum_{\sigma \in \mathbf{S}_q} b_\sigma * (\gamma_h + \gamma_p)^{* \sigma} \right] \Lambda^k \leq 0.$$

Минимизация ведется по переменным a^0, \dots, a^d, T . Данную задачу нелинейного программирования будем решать с использоавнием метода SQP, причем все необходимые производные были найдены с использованием результатов предыдущей главы.

3.3 Численный эксперимент

Конкретизируем задачу, рассмотренную в предыдущем пункте. Пусть требуется посадить БПЛА на платформу, движущуюся по кругу радиусом 2 километра со скоростью 54 км/ч. В момент приземления скорость БПЛА должна быть сонаправлена со скоростью платформы и равняться по величине заданной посадочной скорости. В соответствии с данной постановкой, примем следующие параметры задачи указанные в таблице 1. На рисунках 1 и 2 представлена визуализация процесса решения данной

Таблица 1: Параметры задачи

Параметр	Значение (диапазон)	Единицы
C_u^-, C_u^+	0, 3	g
C_h^-, C_h^+	0, 5000	km
C_v^-, C_v^+	216, 720	km/h
v_l	270	km/h
R	2	km
ω	$54/R$	h^{-1}
$\chi(0)$	[2, 10, 10]	km
$\dot{\chi}(0)$	[0, 270, -36]	km/h
$\chi(T)$	[0, $R \sin(\omega T)$, $R \cos(\omega T)$]	km
$\dot{\chi}(T)$	[0, $v_l \cos(\omega T)$, $-v_l \sin(\omega T)$]	km/h
$\ddot{\chi}(0), \ddot{\chi}(T)$	[0, 0, 0]	km/h^2
k for Λ^k	120	—
deg χ	14	—

задачи с использованием построенного метода и алгоритма SQP. Решение заняло 260 шагов, каждый шаг соответствует серой траектории на визуализации. Найденное минимальное значение финального времени — 82.06 секунды, степень полиномиальной траектории — 14, соответственно количество векторных параметров a^i — 9, количество скалярных переменных по которым велась оптимизация — 28. Все ограничения для найденной квазиоптимальной траектории были выполнены.

Отдельный интерес представляет собой изучение зависимости оптимального значения финального времени от количества векторных параметров. Для второго эксперимента была взята последовательность из 20 задач, для первой задачи использовался только один векторный коэффициент, а далее с шагом единица добавлялись новые коэффициенты и задача решалась заного, и так далее, пока количество коэффициентов не стало равным 20. Добавление нового коэффициента позволяет улучшить время посадки, но с ростом размерности это улучшение все менее заметно, что можно объяснить, во-первых, сложностью работы на задачах большой размерности, а, во-вторых, наличием точной нижней грани для времени посадки. На рисунке 3 изображена норма управления для каждой из 20

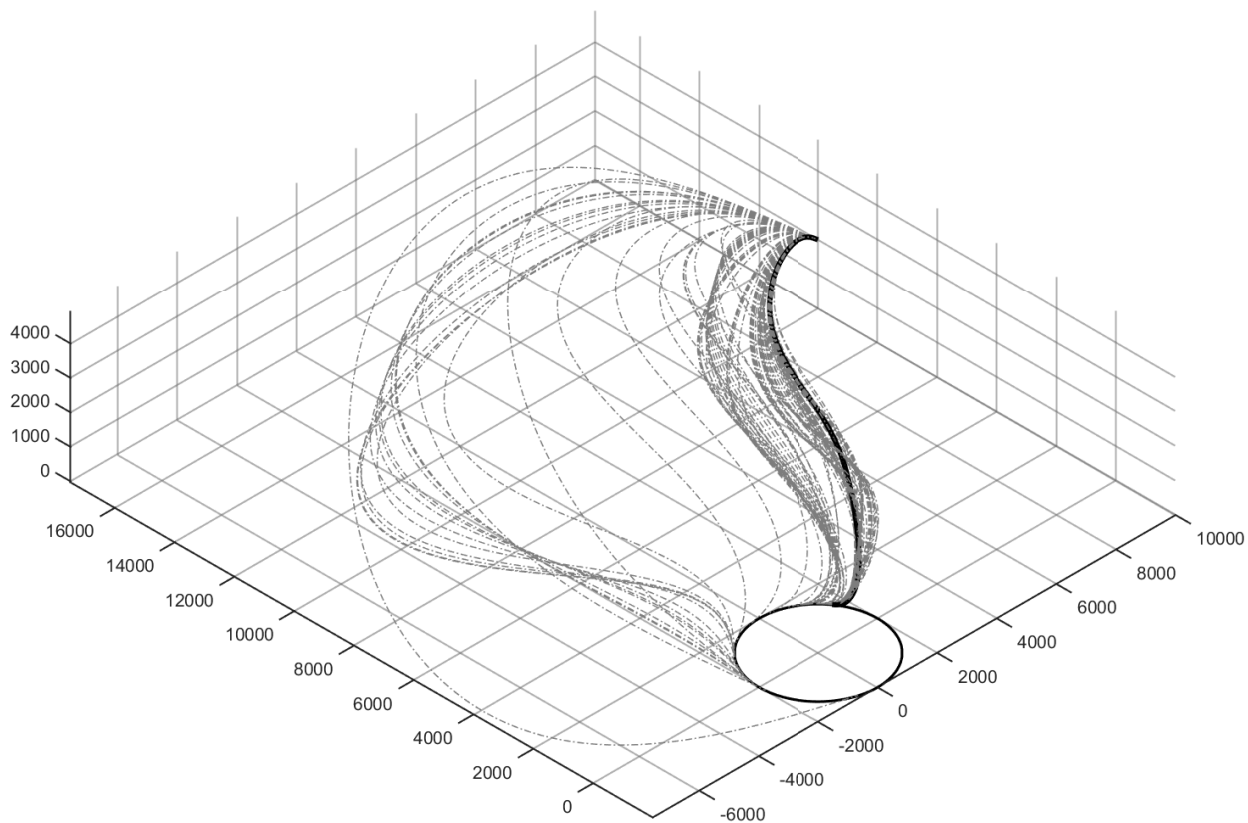


Рис. 1: Визуализация процесса решения. Трехмерный вид. Найденное решение отмечено жирной черной линией. Серыми линиями отмечены траектории, соответствующие шагам алгоритма SQP.

задач, причем, чем более красная линия, тем данное управление оптимальнее по временным затратам. На рисунке 4 указана зависимость найденного наискорейшего времени посадки от количества векторных коэффициентов. Как видно исходя из данного графика, нет особого смысла брать более 10 коэффициентов, что соответствует полиномиальной траектории 15 степени.

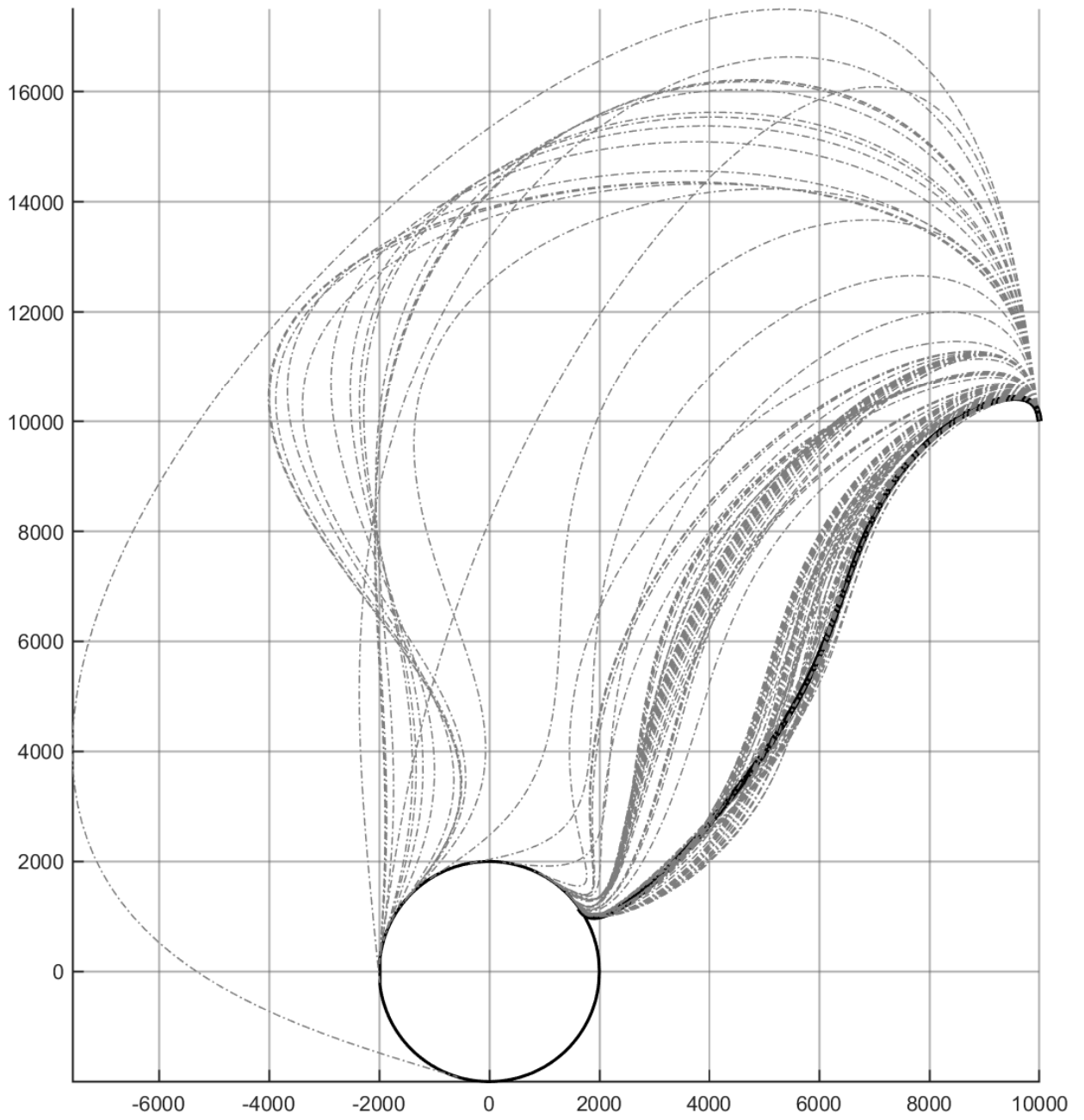


Рис. 2: Визуализация процесса решения. Вид сверху.

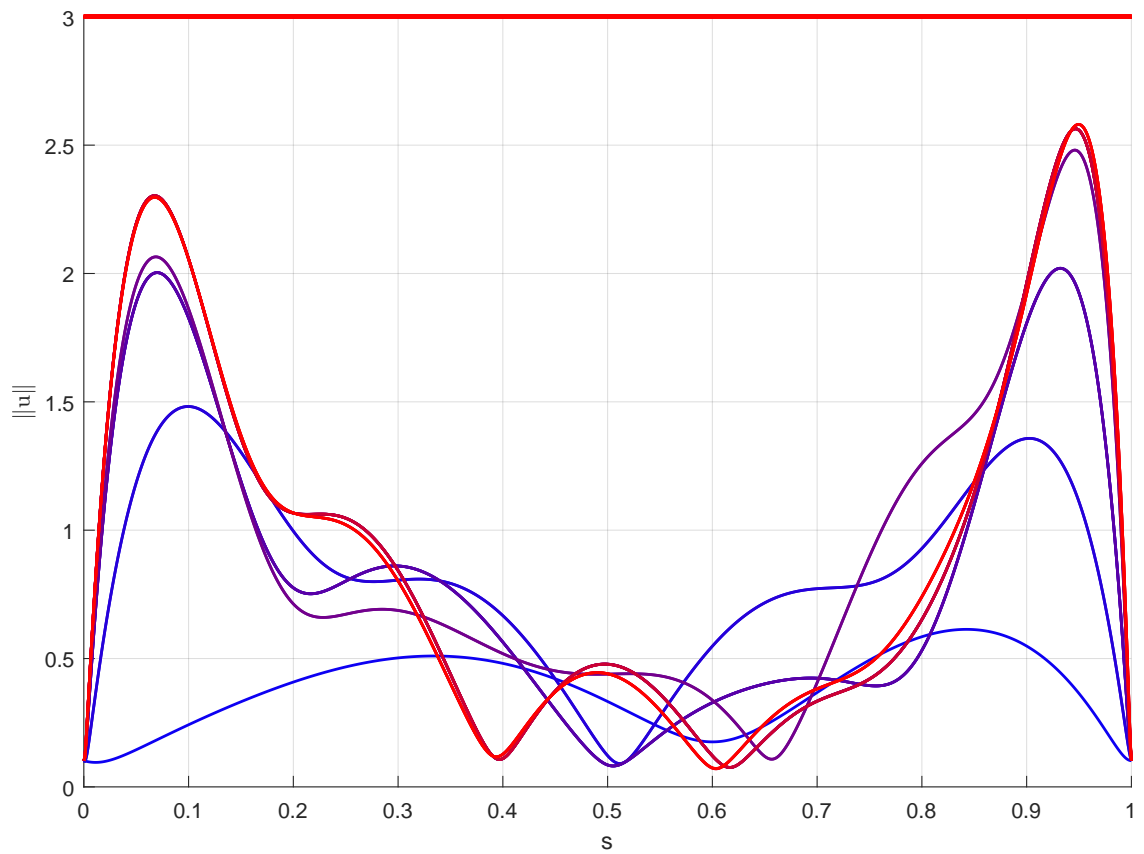


Рис. 3: Норма управления в зависимости от нормализованного времени s . Чем краснее линия, тем оптимальнее управление.

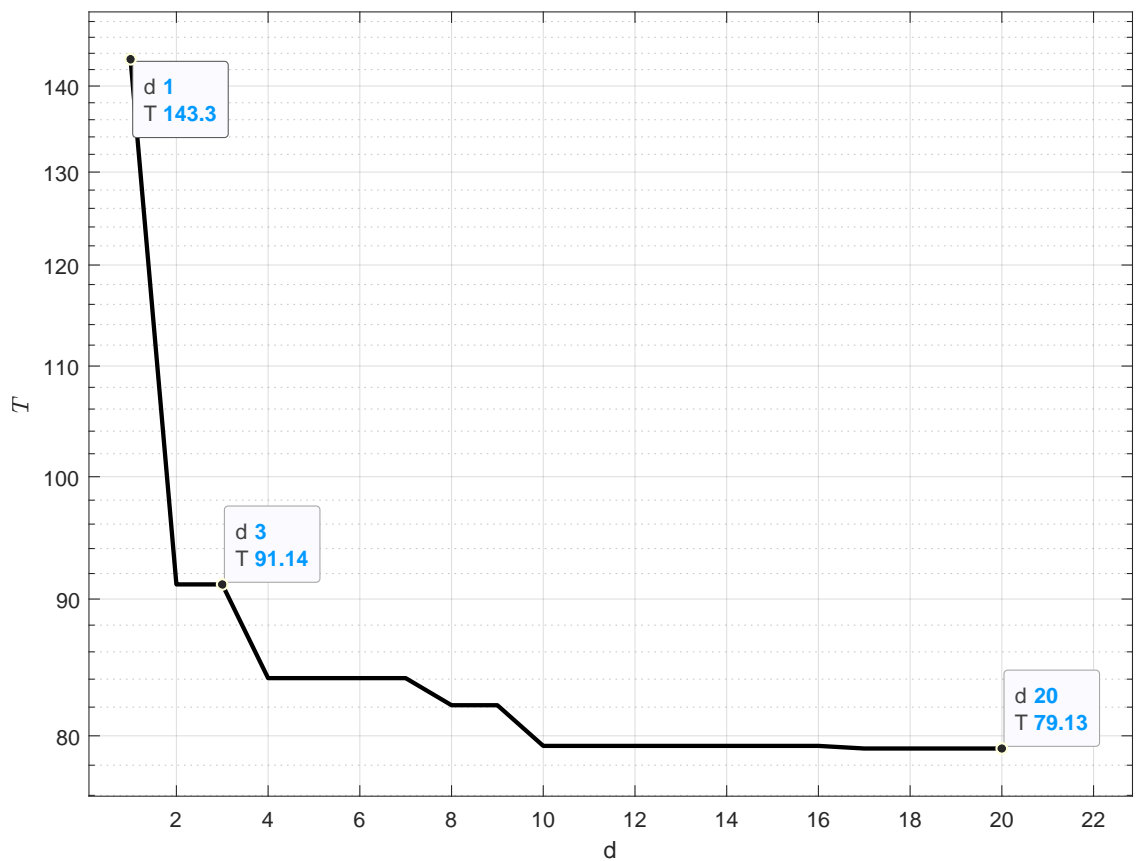


Рис. 4: Зависимость времени посадки от количества коэффициентов a^i .

Заключение

В данной работе был предложен метод сведения задач оптимального управления, представленных в форме дифференциального включения с заданным граничным включением, к задаче нелинейного программирования специального типа, причем метод сведения гарантирует допустимость траектории при выполнении ограничений задачи нелинейного программирования в конечномерном пространстве. В этом заключается основное преимущество метода, в отличие от методов типа коллокаций, когда рассматриваются интерполирующие функции заданные через промежуточные точки, в предложенном методе оптимизация ведется по коэффициентам полинома, который удовлетворяет граничному включению — это позволило нам указать достаточные конечномерные условия допустимости траектории. Как уже было сказано, параметризация с помощью полинома Эрмита обеспечивает точное выполнение граничных условий на траектории конструктивным образом. Представленный метод подходит для случаев с невыпуклыми ограничениями и позволяет выделить выпуклую часть отдельно. Если множество дифференциального включения не представимо в виде пересечения множества с алгебраической границей множества выпуклого, то можно аппроксимировать это множество так, чтобы оно допускало такое представление, от этой аппроксимации, однако, будет зависеть сохранится ли достаточность ограничений задачи конечномерного программирования для допустимости траектории в терминах исходной задачи. Этот вопрос резонно решать для каждой задачи отдельно, так как его общее рассмотрение может быть достаточно громоздко. Изложенный метод проиллюстрирован на примере задачи посадки летательного аппарата на движущуюся по заданному закону платформу. Эта задача является достаточно сложной чтобы её решение с применением принципа максимума Понтрягина было крайне затруднено, но, с другой стороны, данная задача хорошо иллюстрирует подход, когда не требуется находить экстремальное решение, а нужно лишь получить возможность уменьшать время посадки до приемлемого значения посредством добавления дополнительных коэффициентов и осуществления шагов методами наподобие SQP. Показано, что в

рамках данной задачи и при заданных параметрах не имеет смысла строить полиномиальную траекторию выше 15 степени, так как это не влечет практически никаких преимуществ по результирующему времени посадки, и, напротив, может затруднить сходимость.

Список литературы

- [1] Benson, D. A., Huntington, G. T. , Thorvaldsen, T. P. , Rao, A. V.: Direct Trajectory Optimization and Costate Estimation via an Orthogonal Collocation Method. *J. Guid. Control Dyn.* **6**, 1435–1440 (2006)
- [2] Mehne, H. H., Borzabadi, A. H.: A Numerical Method for Solving Optimal Control Problems Using State Parametrization. *Numer. Algorithm.* **2**, 165–169 (2006)
- [3] Kafash, B., Delavarkhalafi, A., Karbassi, S. M.: Application of Chebyshev Polynomials to Derive Efficient Algorithms for the Solution of Optimal Control Problems. *Sci. Iran.* **3**, 795–805 (2012)
- [4] Seywald, H.: Trajectory Optimization Based on Differential Inclusion. *J. Guid. Control Dyn.* **3**, 480–487 (1994)
- [5] Conway, B. A., Larson, K. M.: Collocation Versus Differential Inclusion in Direct Optimization. *J. Guid. Control Dyn.* **5**, 780–785 (1998)
- [6] Hermite, M. Ch., Borchardt, M.: Sur la formule d’interpolation de Lagrange. *J. Reine Angew. Math.* **84**, 70–79 (1878)
- [7] Uteshev, A. Yu., Tamasyan, G. Sh.: On the Problem of Polynomial Interpolation with Multiple Nodes (In Russian). *Vestnik of Saint Petersburg University. Ser. 10.* **3**, 76–85 (2010)
- [8] Lane, J. M., Riesenfeld, R. F.: Bounds on a Polynomial. *BIT Numer. Math.* **1**, 112–117 (1981)
- [9] Rivlin, T. J.: Bounds on a Polynomial. *J. Res. Natl. Bur. Stand. Sec. B: Math.* **1**, 47–54 (1970)

- [10] Nocedal, J., Wright, S. J.: Numerical Optimization, Second Edition. Springer Series in Operations Research, Springer Verlag, New York (2006)
- [11] Waltz, R. A., Morales, J. L., Nocedal, J., Orban, D.: An Interior Algorithm for Nonlinear Optimization that Combines Line Search and Trust Region Steps. Math. Program. **107**, 391–408 (2006)
- [12] Lorentz, G. G.: Bernstein Polynomials. University of Toronto Press, Toronto (1953)
- [13] Kanatnikov, A. N., Shmagina, E. A.: The Problem of Terminal Control of Aircraft Movement (In Russian). In: Emelyanov, S. V., Korovina, S. K. (eds.) Nonlinear Dynamics and Control, vol. 7, pp. 79-94. Fizmatlit, Moscow (2010)