

Расширенный подход к решению задачи оптимального управления в классе реализуемых функций управления

А.И. Дивеев, А.Д. Барабаш

Аннотация— В статье рассматривается актуальная проблема реализации решений классической задачи оптимального управления, применяемая в автоматизированных системах реального времени. Общеизвестно, что прямое применение полученной в рамках классического подхода функции управления, зависящей от времени, неэффективно для реальных объектов из-за разомкнутого характера контура управления и чувствительности к возмущениям. Предлагается инновационный метод, основанный на концепции расширенной модели объекта управления. Ключевая идея метода заключается в первоначальном синтезе универсальной системы стабилизации движения объекта, способной обеспечить точное следование любой заданной траектории в пространстве состояний из широкого класса. Эта система затем интегрируется в объект управления. Для эффективного и автоматического синтеза универсальной системы стабилизации движения применяется метод машинного обучения – символьная регрессия, позволяющая получить компактные и аналитические выражения для управляющих воздействий. В работе детально изложено применение предложенного подхода к сложной задаче оптимального управления квадрокоптером, демонстрируя не только теоретическую возможность, но и практическую реализуемость найденного решения, а также его высокую робастность по отношению к различным начальным возмущениям. Представлены результаты вычислительных экспериментов, подтверждающие эффективность подхода.

Ключевые слова— Квадрокоптер, Машинное обучение, Оптимальное управление, Символьная регрессия, Стабилизация

1. ВВЕДЕНИЕ

Проблема синтеза и практической реализации оптимальных систем управления является одной из центральных в современной теории управления и робототехники [1]. Классическая постановка задачи оптимального управления, как правило, предполагает нахождение такой функции управления $u(t)$, которая минимизирует некоторый функционал качества, переводя систему из заданного начального состояния в желаемое конечное. Однако полученное в результате решение $u(t)$, зависящее исключительно от времени, не учитывает текущие возмущения и неточности модели объекта [2]. Это приводит к формированию разомкнутой системы управления, которая крайне чувствительна к

внешним воздействиям и не способна обеспечить требуемую точность движения реального объекта по оптимальной траектории [3]. Для таких динамически сложных и подверженных влиянию внешней среды объектов, как квадрокоптеры, эта проблема становится особенно острой, поскольку малейшие отклонения от расчетной траектории могут привести к потере устойчивости или аварии [4, 10, 11].

Для решения данной фундаментальной проблемы в настоящей работе предлагается использовать расширенный подход, основанный на концепции расширенной модели объекта управления [5, 9]. Этот подход позволяет преобразовать задачу оптимального управления таким образом, чтобы ее решение стало непосредственно реализуемым на реальном объекте. Ключевым элементом предлагаемого метода является синтез универсальной системы стабилизации движения. В отличие от традиционных систем стабилизации, ориентированных на поддержание равновесия, данная система способна обеспечить точное следование объекта любой заданной траектории в пространстве состояний из достаточно широкого класса, компенсируя при этом неизбежные возмущения и неопределенности [6].

Универсальная система стабилизации движения интегрируется непосредственно в модель объекта управления, преобразуя ее в объект, который способен эффективно "отслеживать" желаемую траекторию. Затем к этому стабилизированному объекту добавляется эталонная модель, которая генерирует оптимальную траекторию. Таким образом, формируется расширенная модель, для которой задача оптимального управления решается уже в классической постановке, но найденное управление $v(t)$ воздействует на эталонную модель, а не напрямую на реальный объект. Это позволяет добиться замкнутого контура управления и значительно повысить робастность всей системы.

Для автоматического синтеза столь сложной и универсальной системы стабилизации применяются методы машинного обучения, в частности, символьная регрессия [7, 8]. Выбор символьной регрессии обусловлен ее уникальной способностью генерировать не просто числовые параметры, а явные аналитические выражения для управляющих законов. Это критически важно для практической реализации, так как позволяет

получить компактные и вычислительно эффективные формулы, пригодные для внедрения в бортовые системы управления квадрокоптерами.

Целью данной статьи является детальное изложение теоретических основ и практического применения расширенного подхода к задаче оптимального управления квадрокоптером. Будет продемонстрировано, как интеграция универсальной системы стабилизации, синтезированной с помощью символьной регрессии, позволяет квадрокоптеру эффективно и точно следовать сложным оптимальным траекториям, даже при наличии начальных возмущений. Результаты вычислительных экспериментов послужат убедительным доказательством реализуемости и перспективности предложенного метода.

II. ПОСТАНОВКА ЗАДАЧИ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ ДЛЯ ПОЛНОГО ЕЕ РЕШЕНИЯ

Традиционная постановка задачи оптимального управления формулируется как нахождение управляющего воздействия $u(t) \in U$,

минимизирующего функционал качества

$$J = \int_{t_0}^{t_f} L(x(t), u(t), t) dt \rightarrow \min \quad (1)$$

для системы, описываемой нелинейными дифференциальными уравнениями состояния:

$$\dot{x} = f(x, u, t), x(t_0) = x^0, \quad (2)$$

где $x \in \mathbb{R}^n$ — вектор состояния объекта, $u \in \mathbb{R}^m$ — вектор управления, t_0 и t_f — начальный и конечный моменты времени, L — функция Лагранжа. Как уже упоминалось, решение данной задачи в классической постановке приводит к разомкнутому контуру управления, что делает ее непрактичной для реальных систем [1].

Для обеспечения реализуемости и робастности решения предлагается метод расширенной модели объекта управления [5]. Этот подход трансформирует исходную задачу оптимального управления в новую, для которой может быть найдено решение, пригодное для непосредственной реализации на физическом объекте.

III. РАСШИРЕННАЯ МОДЕЛЬ

В первую очередь производится интеграция универсальной системы стабилизации в модель объекта. Для исходной динамической системы (1) предварительно синтезируется универсальная система стабилизации движения. Эта система обеспечивает движение объекта $x(t)$ по любой заданной эталонной траектории $x^*(t)$ из некоторого широкого класса. Определяется математическая модель объекта управления (1). Исследованы отдельные программы, реализующие функции управления.

$$V = \{v^{*1}(t), \dots, v^{*M}(t)\}. \quad (3)$$

Определены начальные точки состояний

$$X_0 = \{x^{0,1}, \dots, x^{0,K}\}. \quad (4)$$

Любое программное управление, введенное в правую часть модели (1) вместо вектора свободного управления, генерируется программная траектория как частное решение системы дифференциальных уравнений из заданного начального состояния (3),

$$\dot{x}^{*j}(t, x^0) = f(x(t, x^0), v^{*j}(t)), \quad j = 1, \dots, M. \quad (5)$$

Требуется синтезировать систему управления, обеспечивающую стабилизацию движения по всем программным траекториям при любых начальных условиях, с минимизацией заданного критерия качества.

$$J_1 = \sum_{i=0}^K \sum_{j=1}^M \int_0^{t_i} \|x^{*j}(t) - x(t, x^{0,i})\| dt \rightarrow \min_{u \in U}, \quad (6)$$

где $x^{*j}(t)$ — заданная траектория (5), порождаемая управляющим воздействием (3).

Построение системы стабилизации движения осуществляется на основе функции управления вида:

$$u = h(x^{*j}(t) - x(t)) \in U \quad (7)$$

Управляющий закон (7) выводится методами символьной регрессии. После синтеза функциональной зависимости производится её тестирование в рамках расширенной модели управляемого объекта

$$\begin{aligned} \dot{x}^* &= f(x^*, u), \\ \dot{x} &= f(x, h(x^* - x)), \end{aligned} \quad (8)$$

где первое уравнение является эталонной моделью.

Задача оптимального управления теперь формулируется для нахождения управления $v(t)$, минимизирующего тот же функционал качества J , но уже для расширенной системы. Таким образом, формально задача решается в классической постановке, однако ее физический смысл кардинально меняется. Найденное оптимальное управление $v(t)$ воздействует на эталонную модель, которая генерирует желаемую оптимальную траекторию $x^*(t)$. Благодаря встроенной универсальной системе стабилизации, реальный объект $x(t)$ точно следует этой траектории $x^*(t)$. Этот подход позволяет эффективно решить проблему реализуемости, поскольку $v(t)$ не зависит от текущего состояния реального объекта и может быть вычислено заранее, а стабилизационный контур обеспечивает необходимую корректировку в реальном времени.

IV. СИМВОЛЬНАЯ РЕГРЕССИЯ ДЛЯ СИНТЕЗА СИСТЕМЫ СТАБИЛИЗАЦИИ ДВИЖЕНИЯ

Синтез универсальной системы стабилизации представляет собой сложную задачу, поскольку она должна обеспечить устойчивое следование объекта широкому спектру траекторий при наличии возмущений. Традиционные аналитические методы

синтез часто оказываются слишком трудоемкими или неэффективными для нелинейных систем большой размерности. В данной работе для автоматизированного синтеза применяется машинное обучение, а именно – символьная регрессия [7, 8].

Символьная регрессия (Symbolic Regression, SR) – это мощный парадигма машинного обучения, которая отличается от параметрической регрессии тем, что ищет не просто коэффициенты для заданной формы функции, а саму математическую формулу (структуру) функции, наилучшим образом описывающую взаимосвязь в данных. Это достигается за счет применения эволюционных алгоритмов, таких как генетическое программирование (GP) [12] или картезианское генетическое программирование (CGP). Эти методы оперируют "популяциями" математических выражений, которые подвергаются процессам отбора, скрещивания (комбинации) и мутации, имитируя естественную эволюцию, чтобы найти наиболее "приспособленную" формулу, минимизирующую ошибку предсказания. Реализовать синтез управляющего устройства с выходным сигналом, зависящим от текущего состояния системы

$$u = h(x^* - x) \in U \quad (14)$$

используется численный метод символьной регрессии. В основе подхода лежит представление математических функций специальным кодом с последующим поиском оптимального решения в кодовом пространстве. Разработан усовершенствованный метод символьной регрессии с использованием вариационного универсального кода. Для кодирования математического выражения в вариационном универсальном коде используется алфавит элементарных функций.

Алфавит содержит как функции без параметров, так и необходимые аргументы для математических выражений

$$F_0 = \{f_{0,1} = x_1, \dots, f_{0,n} = x_n, f_{0,n+1} = q_1, \dots, f_{0,n+p} = q_p\}, \quad (15)$$

где $x_1 - x_n$ – переменные, $q_1 - q_n$ – параметры. Набор функций с одним аргументом

$$F_1 = \{f_{1,1}(z) = z, f_{1,2}(z), \dots, f_{1,W}(z)\} \quad (16)$$

Набор функций с двумя аргументами

$$F_2 = \{f_{2,1}(z_1, z_2), \dots, f_{2,V}(z_1, z_2)\}. \quad (17)$$

Кодирование математических выражений осуществляется путём представления каждой элементарной функции в виде пары чисел: первое число определяет количество принимаемых аргументов, второе — идентификатор функции в группе операторов с одинаковой аргументностью.

Унарные функции и переменные выражения кодируются линейно, сохраняя их исходную последовательность в выражении, в то время как бинарные операции записываются в префиксной форме

- сначала идентификатор операции, затем коды её операндов.

Для наглядности рассмотрим конкретный случай. Предположим, имеется математическое выражение:

$$y = \sin(q_1 x_1 + q_2) + \sqrt{x_1^2 + x_2^2} \quad (18)$$

Для представления примера математического выражения можно ограничиться следующим набором базовых функций:

$$\begin{aligned} F_0 &= \{f_{0,1} = x_1, f_{0,2} = x_2, f_{0,3} = q_1, f_{0,4} = q_2\} \\ F_1 &= \{f_{1,1}(z) = z, f_{1,2}(z) = z^2, f_{1,3}(z) = \sin(z), f_{1,4} = \sqrt{z}\} \\ F_2 &= \{f_{2,1}(z_1, z_2) = z_1 + z_2, f_{2,2} = z_1 \cdot z_2\} \end{aligned}$$

Запись выражения (18) в терминах простейших математических операций выглядит следующим образом:

$$y = f_{2,1}(f_{1,3}(f_{2,1}(f_{2,2}(f_{0,3}, f_{0,1}), f_{0,4})), f_{1,4}(f_{2,1}(f_{1,2}(f_{0,1}), f_{1,2}(f_{0,2}))). \quad (19)$$

Универсальный код функции (19) имеет следующий вид (20)

$$y^* = \left(\begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 3 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \end{bmatrix}, \begin{bmatrix} 0 \\ 3 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 4 \end{bmatrix} \right). \quad (20)$$

Числовые значения векторов в коде (20) соответствуют порядковым номерам базовых функций в выражении (19). Для верификации кода применяется индекс элемента k , рассчитываемый по формуле:

$$T(k) = T(k-1) + a_1(k) - 1 \quad (21)$$

где $T(k-1)$ — индекс предыдущего элемента, а $a_1(k)$ - первая компонента кода элементарной функции, соответствующая элементу k .

Для вычисления индекса первого элемента мы принимаем, что индекс предыдущего или "нулевого" элемента $T(0)$ равен нулю.

Используется Критерии Корректности Универсального Кода: универсальный код считается корректным при соблюдении двух условий:

Индекс любого элемента, кроме последнего, должен быть больше нуля (>0).

Индекс последнего элемента всегда равен нулю ($=0$).

Индекс элемента указывает на минимальное количество последующих элементов, которые должны идти за текущим.

Индекс элемента также позволяет находить подвыражения внутри более крупного выражения. Если вы выбираете любой элемент как на начало подвыражения, то его конец будет обозначен первым последующим элементом, чей индекс равен нулю.

Для вычисления значения математического выражения мы начинаем, инициализируя счётчик аргументов $j = 1$. Далее мы последовательно проходим по всем элементам выражения в обратном порядке, начиная с последнего ($k = N$) и двигаясь к первому ($k = 1$). Если первая составляющая кода текущего элемента $a_1(k)$ равна нулю, мы увеличиваем счётчик аргументов ($j \leftarrow j + 1$). В этом случае значение аргумента рассматривается как текущее значение математического выражения.

$$r_j = f_{0,a_2} \quad (22)$$

где a_2 - номер аргумента математического выражения. Если первый компонент элемента кода равен единице, то аргумент и полученный результат подставляются на место текущего значения математического выражения.

$$r_j = f_{1,a_2(k)}(r_j), \quad (23)$$

Если первая составляющая элемента кода равна двум, выполняется вычисление соответствующей функции двух аргументов, где в качестве второго аргумента используется текущее значение математического выражения. Полученный результат подставляется на место текущего значения, после чего значение счётчика уменьшается.

$$r_j = f_{2,a_2(k)}(r_j, r_{j-1}), j \leftarrow j - 1. \quad (24)$$

Вычисления выполняются корректно, если при $k = 1, j = 0$ результат вычислений находится в r_0 .

Универсальный код вариации использует принцип малой вариации базового решения. Одно возможное решение кодируется универсальным кодом, а другие возможные решения в исходной совокупности кодируются набором векторов малой вариации. Чтобы увеличить код, мы используем набор кодов в качестве вектора кодов. Каждый код имеет возможную максимальную длину L и действительную длину $L_i \leq L$, $i = 1, \dots, M$, где M - количество кодов или размерность кодового вектора. После вычисления каждого кода его результат помещается в набор аргументов и может быть использован в последующих вычислениях.

Небольшой вектор вариаций содержит пять компонентов

$$w = [w_1 \ w_2 \ w_3 \ w_4 \ w_5]^T, \quad (25)$$

где w_1 - тип вариации, w_2 - кодовый номер, w_3 - позиция в коде, w_2, w_4, w_5 - дополнительные компоненты вектора малых вариаций.

В универсальном коде для обозначения вариации типа используются. Тип $w_1 = 0$ соответствует изменению второго элемента кода

$$a_{w_2,2}(w_3) \leftarrow w_4. \quad (26)$$

Тип $w_1 = 1$ соответствует удалению функции с одним или двумя аргументами. Если $a_{w_2,1}(w_3) = 1$, то

функция с одним аргументом удаляется. Если $a_{w_2,1}(w_3) = 2$, то функция с двумя аргументами удаляется. При удалении функции с двумя аргументами удаляется следующее подвыражение, начинающееся с элемента, следующего за элементом функции w_3 .

Тип $w_1 = 2$ соответствует вставке функции с одним аргументом. Если $L_{w_2} < L$, то все элементы после позиции w_3 перемещаются вправо на одну позицию, а затем вставляется код функции с одним аргументом,

$$a_{w_2,1}(w_3) = 1, a_{w_2,2}(w_3) = w_4. \quad (27)$$

Тип $w_1 = 3$ соответствует вставке функции с двумя аргументами. Если $L_{w_2} < L - 1$, то все элементы после позиции w_3 перемещаются на две позиции вправо, а затем в позицию w_3 вставляется код функции с двумя аргументами, а в позицию $w_3 + 1$ вставляется код аргумента математического выражения,

$$a_{w_2,1}(w_3) \leftarrow 2, a_{w_2,2}(w_3) \leftarrow w_4, a_{w_2,1}(w_3 + 1) \leftarrow 0, a_{w_2,2}(w_3 + 1) \leftarrow w_5. \quad (28)$$

Каждое возможное решение, за исключением базового, кодируется набором векторов малых вариаций.

$$W_i = (w^{i,1}, \dots, w^{i,d}), \quad (29)$$

где d - количество векторов малых вариаций в одном наборе или глубина вариаций базового решения.

Для поиска оптимального решения используется вариационный генетический алгоритм. Операции скрещивания и мутации выполняются над наборами векторов малых вариаций. Через несколько поколений базовое решение меняется на наилучшее из возможных, найденное на данный момент. Этот процесс называется сменой эпохи.

V. ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ

Рассмотрим задачу оптимального управления группой из двух роботов с дифференциальным приводом. Математическая модель объектов управления имеет следующий вид

$$\begin{aligned} \dot{x}_1 &= 0.5(u_1 + u_2) \cos(x_3), \\ \dot{x}_2 &= 0.5(u_1 + u_2) \sin(x_3), \\ \dot{x}_3 &= 0.5(u_1 - u_2), \end{aligned} \quad (30)$$

$$\begin{aligned} \dot{x}_4 &= 0.5(u_3 + u_4) \cos(x_3), \\ \dot{x}_5 &= 0.5(u_3 + u_4) \sin(x_3), \\ \dot{x}_6 &= 0.5(u_3 - u_4), \end{aligned}$$

где $x^{(1)} = [x_1 \ x_2 \ x_3]^T$ - вектор пространства состояний первого робота, $x^{(2)} = [x_4 \ x_5 \ x_6]^T$ - вектор пространства состояний второго робота, $u^{(1)} = [u_1 \ u_2]^T$ - элемент управления вектор первого робота, $u^{(2)} = [u_3 \ u_4]^T$ - это управляющий вектор второго робота.

Управление роботами ограничено

$$-10 = u^- \leq u_i \leq u^+ = 10, \quad i = 1, 2, 3, 4. \quad (31)$$

Задано исходное состояние

$$x^0 = [0 \ 0 \ 0 \ 10 \ 10 \ \pi]^T. \quad (32)$$

Задано конечное состояние

$$x^f = [10 \ 10 \ 0 \ 0 \ 0 \ \pi]^T. \quad (33)$$

Приведены фазовые ограничения

$$\varphi_i(x^{(j)}) = R_i - \sqrt{(x_{1,i} - x_{1+3(j-1)})^2 + (x_{2,i} - x_{2+3(j-1)})^2} \leq 0, \quad (34)$$

где $i = 1, 2, j = 1, 2, R_1 = R_2 = 2, x_{1,1} = 5, x_{2,1} = 2, x_{1,2} = 5, x_{2,2} = 8$.

Задано условие избежание столкновения

$$\chi(x) = R_0 - \sqrt{(x_1 - x_4)^2 + (x_2 - x_5)^2} \leq 0, \quad (35)$$

где $R_0 = 1, 1$.

Приведен критерий качества. Для численного решения этот критерий может включать проверку фазовых ограничений и достижение конечного состояния.

$$J = t_f + p_1 \sum_{i=1}^2 \sum_{j=1}^2 \int_0^{t_f} \chi(x) + \vartheta(\varphi_i(x^{(j)})) + p_2 \|x^f - x(t_f)\| \rightarrow \min_u, \quad (36)$$

где $p_1 = 4, p_2 = 1, t_f = \max\{t_{f,1}, t_{f,2}\}, t_{f,j}$ - время достижения роботом j конечного состояния,

$$\vartheta(\alpha) = \begin{cases} 1, & \text{если } \alpha > 0 \\ 0 & \text{иначе} \end{cases} \quad (37)$$

Для решения задачи оптимального управления в классе непосредственно реализующих функции управления используется расширенная модель. Первоначально мы синтезируем универсальную систему стабилизации.

Для этой цели используется метод символьной регрессии, вариационный универсальный код. В результате была найдена следующая функция управления

$$u_{i+3(j-1)} = \begin{cases} u^+, & \text{если } u^+ \leq \tilde{u}_{i+3(j-1)} \\ u^-, & \text{если } \tilde{u}_{i+3(j-1)} \leq u^- \\ \tilde{u}_{i+3(j-1)}, & \text{в других сл.} \end{cases} \quad (38)$$

где $i = 1, 2, j = 1, 2$,

$$\tilde{u}_{1+3(j-1)} = \mu(r_{10} + r_9 + \rho_{19}(r_8) + \rho_{17}(r_5) + \operatorname{sgn}(\rho_{17}(r_4)) + r_3^2 + r_2^2 - \Delta_2 + \exp(r_2) + \vartheta(q_2 \Delta_2 \operatorname{sgn}(\Delta_1))), \quad (39)$$

$$\tilde{u}_{2+3(j-1)} = \vartheta(r_{11}) + \operatorname{sgn}(r_{10}) \sqrt{|r_{10}|} + \rho_{19}(r_8) - r_7 + (q_3)^{-1} + \vartheta(\Delta_1), \quad (40)$$

$$r_{10} = r_9 - r_9^3 + r_6^2 + \operatorname{sgn}(r_4) - r_2 + \ln(|q_3 \exp(q_2)|) \Delta_3 \cos(\Delta_1),$$

$$r_9 = r_7 + q_1 + \cos(r_7) + \sin(q_3 \exp(q_2)) + \arctan(q_2 \Delta_2 \operatorname{sgn}(\Delta_1) \Delta_1),$$

$$r_8 = \rho_{18}(r_6) + (r_5 q_2 r_6 \Delta_2 \operatorname{sgn}(\Delta_1 + \tanh(\Delta_3)))^{-1},$$

$$r_7 = \vartheta(r_4) \exp(q_2 \operatorname{sgn}(\Delta_1) \rho_{17}(q_1))$$

$$r_6 = \rho_{16}(r_4) + r_5 + (q_3)^{-1} + \rho_{18}(\Delta_3),$$

$$r_5 = r_4^2 - r_3 \operatorname{sgn}(r_1) \rho_{19}(q_3) \rho_{18}(\Delta_3) \operatorname{sgn}(\Delta_2) \sqrt{|\Delta_2|},$$

$$\begin{aligned} r_4 &= q_3(r_2 - q_2), \\ r_3 &= \Delta_1 + \mu(\Delta_3^3) + \Delta_2^3 + \Delta_1, \\ r_2 &= q_3 \exp(\Delta_3) q_2 + \Delta_2 \Delta_1, \\ r_1 &= q_3(q_1 \Delta_1 + \sin(q_1) + \Delta_2^3), \\ \rho_{17}(\alpha) &= \operatorname{sgn}(\alpha) \ln(|\alpha| + 1), \\ \rho_{18}(\alpha) &= \operatorname{sgn}(\alpha) \exp(\alpha) - 1, \\ \rho_{19}(\alpha) &= \operatorname{sgn}(\alpha) \exp(-|\alpha|), \\ \mu(\alpha) &= \begin{cases} \alpha, & \text{если } |\alpha| \leq 1 \\ \operatorname{sgn}(\alpha), & \text{иначе} \end{cases} \end{aligned}$$

$\Delta_i = x_{i+3(j-1)}^*(t) - x_{i+3(j-1)}(t), i = 1, 2, 3,$
 $x^*(t)$ - оптимальная траектория, $q_1 = 13,84326,$
 $q_2 = 2,27808, q_3 = 1,24121$.

Впоследствии в расширенную модель была включена универсальная система стабилизации (38) вместе с моделью. (30), которая для расширенной модели была эталонной. В общей сложности расширенная модель включала двенадцать уравнений.

Проекция траекторий движения роботов на горизонтальную плоскость из восьми возмущенных начальных состояний показаны на рисунках 1 для первого робота и на рисунке 2 для второго робота. Также приведены оптимальные траектории (точки).

Как видно из результатов экспериментов, траектории движения роботов из возмущенных начальных состояний приближаются к выбранной оптимальной траектории.

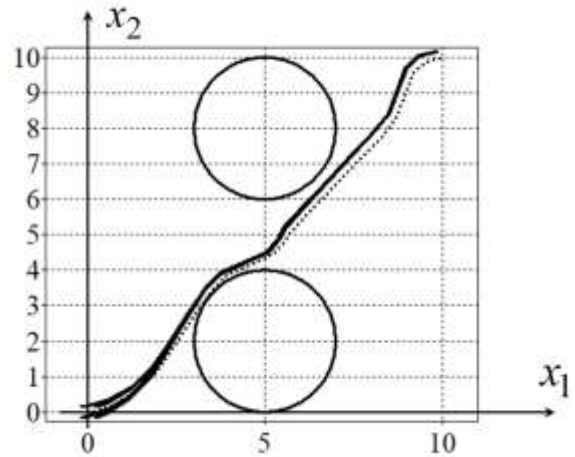


Рис. 1. Проекция траекторий первого робота на горизонтальную плоскость из восьми возмущенных начальных состояний. Оптимальная траектория (точки)

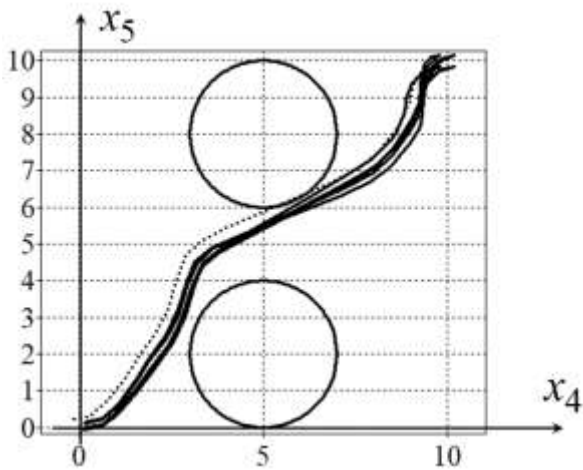


Рис. 2. Проекция траекторий второго робота на горизонтальную плоскость из восьми возмущенных начальных состояний. Оптимальная траектория (точки)

Для проверки качества универсальной системы стабилизации (38) с помощью той же расширенной модели была решена еще одна задача оптимального управления. В новой задаче рассматривались фазовые ограничения другого размера и другого расположения, $R_1 = R_2 = 2.2$, $x_{1,1} = 2$, $x_{2,1} = 6$, $x_{1,2} = 8$, $x_{2,2} = 4$. Роботы стартовали из других начальных состояний, также двигались навстречу друг другу, чтобы поменяться местами. Результаты проведенных экспериментов представлены на рисунках 3 и 4.

Результаты эксперимента показали, что универсальная стабилизация обеспечивает качественное перемещение объекта вблизи оптимальную траекторию, при этом одна и та же система стабилизации движения может быть использована для различных задач оптимального управления этим объектом. Возмущения начальных состояний в экспериментах составили $\pm 0,15$ от всех составляющих вектора состояния.

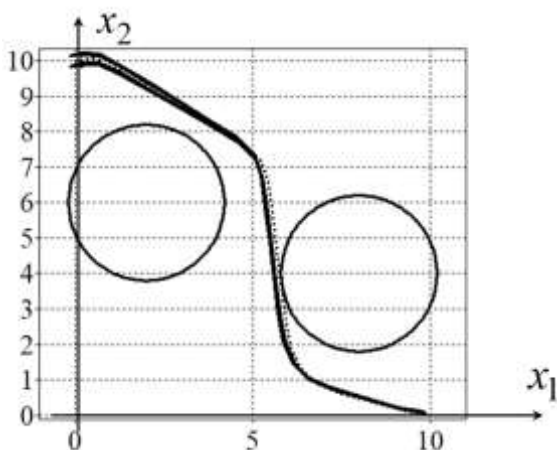


Рис. 3. Проекция траекторий первого робота на горизонтальную плоскость из восьми возмущенных начальных состояний во второй задаче.

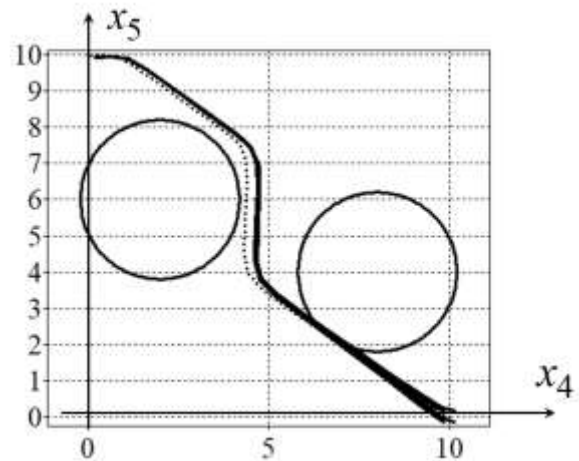


Рис. 4. Проекция траекторий второго робота на горизонтальную плоскость из восьми возмущенных начальных состояний во второй задаче

VI. ВЫВОДЫ

В статье представлен метод расширенной модели для решения задачи оптимального управления в классе реализуемых функций. Метод изначально решает задачу синтеза системы стабилизации движения объекта вблизи заданной траектории. Для решения задачи синтеза используется символьная регрессия. В данной статье представлен новый метод символьной регрессии - вариационный универсальный код. Метод включает в себя два хорошо известных метода символьной регрессии: декартово генетическое программирование и генетическое программирование на основе генетического кода. Для синтеза заданы универсальная система стабилизации движения, множественность траекторий и набор начальных состояний, близких к заданным траекториям.

Расширенная модель представляет собой модель объекта управления с системой стабилизации движения по заданной траектории из множества траекторий определенного класса и эталонной моделью для генерации оптимальной траектории. В результате размерность расширенной модели объекта управления в два раза превышает размерность самой модели. В статье представлено решение двух задач оптимального управления со статическими и динамическими фазовыми ограничениями для группы из двух роботов с дифференциальным приводом. Возмущения в исходных состояниях роботов с универсальной системой стабилизации не существенно изменили траекторию движения. Таким образом, решение задачи оптимального управления роботами возможно установить на реальных объектах управления.

БИБЛИОГРАФИЯ

- [1] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, E. F. Mishchenko, The Mathematical Theory of Optimal Process. Gordon and Breach Science Publishers, New York, London, Paris, Montreux, Tokyo: 1985. L. S. Pontryagin, Selected works, Volume 4. 360 p.
- [2] M. Egerstedt, Motion Planning and Control of Mobile Robots. Ph.D. Thesis, Royal Institute of Technology, Stockholm, Sweden, 2000.

- [3] G. Walsh, D. Tilbury, S. Sastry, R. Murray, J.P. Laumond, Stabilization of trajectories for systems with nonholonomic constraints. *IEEE Trans. Autom. Control* 1994, 39, pp. 216-222.
- [4] A. Samir, A. Hammad, A. Hafez, H. Mansour, Quadcopter Trajectory Tracking Control using State-Feedback Control with Integral Action. *Int. J. Comput. Appl.* 2017, 168, pp. 1-7.
- [5] A.I. Diveev, E.Yu. Shmalko, V.V. Serebrenny, P. Zentay, Fundamentals of Synthesized Optimal Control. *Mathematics* 2021, 9, 21.
- [6] A. Diveev, E. Shmalko, Stabilization System Quality of Synthesized Optimal Control. *Mathematics* 2024, 1,0.
- [7] A. Diveev, E. Shmalko, Machine Learning Control by Symbolic Regression. Springer International Publishing, Cham, 2021.
- [8] A. Diveev, E. Shmalko, Machine-Made Synthesis of Stabilization System by Modified Cartesian Genetic Programming// *IEEE Transactions on Cybernetics*, 2022, 52(7), pp. 6627-6637.
- [9] A. Diveev, E. Sofronova, Universal Stabilization System for Control Object Motion along the Optimal Trajectory. *Mathematics* 2023, 11, 3556.
- [10] Zhao, Han, Su, Guo, Yang. Anti-collision Trajectory Planning and Tracking Control based on MPC and Fuzzy PID Algorithm. *Proceedings of the 2020 4th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, Hangzhou, China, December 18-20, 2020, pp. 613-618.
- [11] Tingting Su, Xu Liang, Guangping He, Quanliang Zhao, Lei Zhao. Robust Trajectory Tracking of Delta Parallel Robot Using Sliding Mode Control. *Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI)* December 6-9 2019, Xiamen, China, 508-512
- [12] Koza, J.R. Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, 1992, 819 p.

Статья получена 7 июня 2025

Дивеев А.И - д.т.н., профессор главный научный сотрудник
Федерального государственного учреждения Федерального
исследовательского центра «Информатика и управление» Российской
академии наук (ФИЦИУ РАН), руководитель отдела 55 Управление
робототехническими устройствами, директор роботцентра ФИЦИУ
РАН. (email: aidiveev@mail.ru)

Барабаш А.Д. - Аспирант Федерального государственного
учреждения Федерального исследовательского центра «Информатика
и управление» Российской академии наук (ФИЦИУ РАН). (email:
artew44@gmail.com)

An extended approach to solving the optimal control problem in the class of implemented control functions

A.I. Diveev, A.D. Barabash

Abstract— The article discusses the actual problem of implementing solutions to the classical optimal control problem used in automated real-time systems. It is well known that the direct application of the time-dependent control function obtained within the framework of the classical approach is ineffective for real objects due to the open nature of the control loop and sensitivity to disturbances. An innovative method based on the concept of an extended management object model is proposed. The key idea of the method is the initial synthesis of a universal object motion stabilization system capable of accurately following any given trajectory in a wide class of states. This system is then integrated into the management facility. For the effective and automatic synthesis of a universal motion stabilization system, a machine learning method is used – symbolic regression, which makes it possible to obtain compact and analytical expressions for control actions. The paper describes in detail the application of the proposed approach to the complex problem of optimal control of a quadcopter, demonstrating not only the theoretical possibility, but also the practical feasibility of the solution found, as well as its high robustness with respect to various initial disturbances. The results of computational experiments confirming the effectiveness of the approach are presented.

Keywords — Quadcopter, Machine learning, Optimal control, Symbolic regression, Stabilization

REFERENCES

- [1] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, E. F. Mishchenko, The Mathematical Theory of Optimal Process. Gordon and Breach Science Publishers, New York, London, Paris, Montreux, Tokyo: 1985. L. S. Pontryagin, Selected works, Volume 4. 360 p.
- [2] M. Egerstedt, Motion Planning and Control of Mobile Robots. Ph.D. Thesis, Royal Institute of Technology, Stockholm, Sweden, 2000.
- [3] G. Walsh, D. Tilbury, S. Sastry, R. Murray, J.P. Laumond, Stabilization of trajectories for systems with nonholonomic constraints. IEEE Trans. Autom. Control 1994, 39, pp. 216–222.
- [4] A. Samir, A. Hammad, A. Hafez, H. Mansour, Quadcopter Trajectory Tracking Control using State-Feedback Control with Integral Action. Int. J. Comput. Appl. 2017, 168, pp. 1–7.
- [5] A.I. Diveev, E.Yu. Shmalko, V.V. Serebrenny, P. Zentay, Fundamentals of Synthesized Optimal Control. Mathematics 2021, 9, 21.
- [6] A. Diveev, E. Shmalko, Stabilization System Quality of Synthesized Optimal Control. Mathematics 2024, 1,0.
- [7] A. Diveev, E. Shmalko, Machine Learning Control by Symbolic Regression. Springer International Publishing, Cham, 2021.
- [8] A. Diveev, E. Shmalko, Machine-Made Synthesis of Stabilization System by Modified Cartesian Genetic Programming//IEEE Transactions on Cybernetics, 2022, 52(7), pp. 6627–6637.
- [9] A. Diveev, E. Sofronova, Universal Stabilization System for Control Object Motion along the Optimal Trajectory. Mathematics 2023, 11, 3556.
- [10] Zhao, Han, Su, Guo, Yang. Anti-collision Trajectory Planning and Tracking Control based on MPC and Fuzzy PID Algorithm. Proceedings of the 2020 4th CAA International Conference on Vehicular Control and Intelligence (CVCI), Hangzhou, China, December 18-20, 2020, pp. 613-618.
- [11] Tingting Su, Xu Liang, Guangping He, Quanliang Zhao, Lei Zhao. Robust Trajectory Tracking of Delta Parallel Robot Using Sliding Mode Control. Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI) December 6-9 2019, Xiamen, China, 508-512
- [12] Koza, J.R. Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, 1992, 819 p.