Интеллектуальные системы управления, анализ данных

© 2025 г. И.А. ЧИСТЯКОВ (chistyakov.ivan@yahoo.com)
(Московский государственный университет им. М.В. Ломоносова)

О ГАРАНТИРОВАННОЙ ОЦЕНКЕ ОТКЛОНЕНИЯ ОТ ЦЕЛЕВОГО МНОЖЕСТВА В ЗАДАЧЕ УПРАВЛЕНИЯ ПРИ ОБУЧЕНИИ С ПОДКРЕПЛЕНИЕМ[1]

Рассматривается задача целевого управления объектом, движение которого описывается системой дифференциальных уравнений специального вида, где присутствуют нелинейные члены, зависящие от фазовых переменных. На примере алгоритма Proximal Policy Optimization (PPO) показано, что с помощью обучения с подкреплением можно получить позиционную стратегию управления, решающую задачу приближенно. Эта стратегия далее аппроксимируется кусочно-аффинным управлением, для которого на основе метода динамического программирования строится гарантированная априорная оценка попадания траектории в целевое множество. Для этого осуществляется переход к вспомогательной задаче для кусочно-аффинной системы с помехой и вычисляется кусочно-квадратичная оценка функции цены как приближенное решение уравнения Гамильтона–Якоби–Беллмана.

Ключевые слова: нелинейная динамика, динамическое программирование, принцип сравнения, линеаризация, кусочно-квадратичная функция цены, обучение с подкреплением, алгоритм PPO, множество разрешимости.

DOI: 10.31857/S0005231025010057, **EDN:** JQKKTQ

1. Введение

Рассматривается задача целевого управления на фиксированном конечном интервале времени для нелинейной системы дифференциальных уравнений. Такая задача тесно связана с построением множества разрешимости, содержащего все стартовые позиции, из которых можно решить задачу синтеза управлений. Для аппроксимации этого множества применяются различные методы на основе анализа соответствующего дифференциального включения [1–3] или на основе уравнения Гамильтона–Якоби–Беллмана (ГЯБ) [4–7]. Указанные подходы применимы для широкого класса нелинейных систем, но

[1] Работа выполнена при финансовой поддержке Минобрнауки России в рамках реализации программы Московского центра фундаментальной и прикладной математики по соглашению № 075-15-2022-284.

требуют больших вычислительных затрат. В последнее время активно разрабатываются алгоритмы на основе машинного обучения, которые позволяют как приблизить решение уравнения ГЯБ [8, 9], так и осуществить поиск управления напрямую [10]. Последние, однако, не дают возможность получить какие-либо гарантированные оценки.

В данной работе предлагается понизить вычислительную сложность решения уравнения ГЯБ за счет поиска приближенного решения в классе кусочно-квадратичных функций. Развиваются идеи, изложенные в [11–13]: используется метод, основанный на кусочной линеаризации правых частей дифференциальных уравнений на совокупности симплексов и переходе к задаче управления для системы с кусочно-линейной динамикой и ограниченной помехой (погрешностью линеаризации). Применение принципа сравнения [14, 15] позволяет вывести уравнения на коэффициенты искомой функции цены, нулевое множество уровня которой является внутренней оценкой множества разрешимости исходной нелинейной системы.

Приближенное решение уравнения ГЯБ упомянутым способом сопровождается построением субоптимальной управляющей стратегии. Ранее было предложено искать управление в виде непрерывной кусочно-аффинной функции [13], определяемой значениями в вершинах симплексов разбиения. При этом значения в вершинах следует выбирать таким образом, чтобы минимизировать производную функции цены вдоль траектории движения. Однако с учетом отсутствия гладкости построенной функции цены приходится применять дополнительные эвристики, увеличивающие погрешность метода. В настоящей работе поставлена цель продемонстрировать, что в качестве управлений в вершинах также могут быть использованы результаты других алгоритмов, в частности предлагается применять обучение с подкреплением [16, 17]. Показано, что если выбирать значения управлений на основе нейросетевой модели, то полученная оценка функции цены способна принимать меньшие значения в начальный момент времени, что априорно гарантирует попадание в меньшую окрестность целевого множества.

Отметим, что алгоритмы обучения с подкреплением также подразумевают построение функции цены, которая является оценкой результирующей выгоды из каждой возможной позиции (в данном случае речь идет о расстоянии до целевого множества в конечный момент времени), или ее аналогов. Но даже при удачно подобранном управлении такая оценка не является гарантированной и может быть неточной. В то же время подход, указанный в настоящей работе, позволяет приблизить любую наперед заданную стратегию кусочно-аффинным управлением, для которого полученная оценка будет гарантированной. Это может быть особенно полезно в случае наличия дополнительной помехи, когда вычисления траекторий из различных начальных точек оказывается недостаточно, чтобы оценить все возможные варианты поведения системы.

2. Постановка задачи

Рассмотрим нелинейную систему дифференциальных уравнений:

$$\dot{x} = \mathbf{f}(t,x) + \mathbf{g}(t,x)u, \quad t \in [t_0,t_1],\ x \in \Omega, \tag{1}$$

где Ω – компактное множество в пространстве $\mathbb{R}^{n_x}$, достаточно большое, чтобы все рассматриваемые траектории системы (1) оставались в Ω при $t \in \in [t_0,t_1]$; будем полагать, что границей Ω является многогранник. Нелинейные вектор-функция $\mathbf{f}(t,x)$ и матричная функция $\mathbf{g}(t,x) \in \mathbb{R}^{n_x \times n_u}$ непрерывны по t и дважды непрерывно дифференцируемы по x. Начальный и конечный моменты времени t_0, t_1 фиксированы. В каждый момент времени вектор управления u должен принадлежать компактному выпуклому множеству $\mathcal{P}$:

$$u \in \mathcal{P} \subset \mathbb{R}^{n_u}. \tag{2}$$

Требуется построить непрерывную управляющую стратегию в позиционной форме $u = u(t,x)$, которая переводит систему (1) из заданной точки x_0 в момент времени t_0 в как можно меньшую окрестность компактного целевого множества $\mathcal{X}_1 \subset \Omega$ в момент времени t_1. Далее через $u(\cdot)$ будем обозначать позиционные управления. Таким образом, должно выполняться

$$x(t_1;t_0,x_0)|_{u(\cdot)} \in \mathcal{X}_1 + B_\varepsilon(0),$$

где $x(t_1;t_0,x_0)|_{u(\cdot)}$ – точка траектории системы в момент времени t_1, выпущенной в момент t_0 из точки x_0 при замыкании этой системы управлением $u(\cdot)$; $B_\varepsilon(0)$ – шар радиуса ε с центром в нуле, а значение $\varepsilon \geqslant 0$ необходимо минимизировать. Будем также считать, что целевое множество представимо в виде $\mathcal{X}_1 = \{x \in \Omega : \phi_{\mathcal{X}_1}(x) \leqslant 0\}$, где $\phi_{\mathcal{X}_1}(x)$ – дважды непрерывно дифференцируемая функция.

Кроме того, необходимо построить *множество разрешимости* $\mathcal{W}(t,t_1,\mathcal{X}_1)$ [15], т.е. совокупность всех векторов $x \in \Omega$, для каждого из которых существует управление $u(\cdot)$, удовлетворяющее ограничению (2) и переводящее систему из позиции $\{t,x\}$ $(t \in [t_0,t_1])$ в целевое множество: $x(t_1;t,x)|_{u(\cdot)} \in \mathcal{X}_1$. Однако поскольку задача построения точного множества разрешимости является сложной, далее ограничимся поиском внутренних оценок этого множества.

3. Система с кусочно-аффинной динамикой

Симплексом [18] размерности n с вершинами $x_1, x_2, \ldots, x_{n+1} \in \mathbb{R}^n$ при условии, что векторы $x_2 - x_1, \ldots, x_{n+1} - x_1$ являются линейно независимыми, называется множество

$$S^n = \left\{ \alpha_1 x_1 + \alpha_2 x_2 + \ldots + \alpha_{n+1} x_{n+1} : \alpha_i \geqslant 0,\ \sum_{i=1}^{n+1} \alpha_i = 1 \right\}.$$

При этом вектор *барицентрических координат* $\alpha(x) = (\alpha_1, \ldots, \alpha_{n+1})^{\mathrm{T}}$ однозначно определяет положение любой точки x внутри симплекса. Кроме того,

существует матрица $\tilde{H}$ [11] такая, что барицентрические координаты $\alpha(x)$ линейно выражаются через x: $\alpha = \tilde{H}(x^{\mathrm{T}}, 1)^{\mathrm{T}}$.

Пусть задано разбиение множества Ω на N симплексов $\Omega^{(i)}$. При этом будем считать, что любые два симплекса либо не пересекаются, либо пересекаются только по какой-либо их общей грани размерности меньшей n_x. В практических задачах, имея произвольный набор вершин, конкретное разбиение можно реализовать, например, с помощью триангуляции Делоне [19, 20], которая эффективно вычисляется за счет построения выпуклой оболочки точек в $(n_x + 1)$-мерном пространстве [21].

Далее верхним индексом (i) будем обозначать соответствие вектора, матрицы или функции симплексу $\Omega^{(i)}$. В частности, обозначим вершины симплексов как $g_1^{(i)}, \ldots, g_{n_x+1}^{(i)} \in \mathbb{R}^{n_x}$, где $i = \overline{1, N}$. Отметим, однако, что каждая такая вершина может являться вершиной сразу нескольких симплексов.

В работах [11–13] был предложен способ построения непрерывной кусочно-аффинной аппроксимации правой части системы (1), существенно использующий разбиение множества Ω на симплексы. Было показано, как выбрать матрицы $A^{(i)}$, $B^{(i)}$ и векторы $f^{(i)}$ так, что сразу для всех $u \in \mathcal{P}$ будет справедливо представление

$$\mathbf{f}(t,x) + \mathbf{g}(t,x)u = A^{(i)}(t)x + B^{(i)}(t)u + f^{(i)}(t) + v^{(i)}(t,x,u), \quad x \in \Omega^{(i)}, \tag{3}$$

где $v^{(i)}$ – погрешность локальной линеаризации. Эта погрешность является ограниченной, и для нее существует оценка на основе разложения компонент вектор-функций $\mathbf{f}(t,x)$ и $\mathbf{g}(t,x)$ по формуле Тейлора, не зависящая от конкретного значения x во множестве $\Omega^{(i)}$ и управления. Таким образом, всевозможные значения $v^{(i)}$ можно ограничить некоторым эллипсоидом $\mathcal{Q}^{(i)}(t)$:

$$\begin{gathered} \mathcal{Q}^{(i)}(t) = \mathcal{E}(0, Q^{(i)}(t)) = \{x \in \mathbb{R}^{n_x} : \ \langle x, (Q^{(i)})^{-1} x \rangle \leqslant 1\}, \\ Q^{(i)} = (Q^{(i)})^{\mathrm{T}} > 0. \end{gathered} \tag{4}$$

Замечание 1. Если в системе (1) будет дополнительно присутствовать аддитивный член в виде неизвестной ограниченной функции (помехи), то он также может быть учтен при линеаризации системы за счет увеличения эллипсоидов $\mathcal{Q}^{(i)}(t)$ и смещения их центров.

Удобно перейти к расширенному пространству переменных, где вектор $\tilde{x}$ получается добавлением вспомогательной координаты с фиксированным значением, равным единице: $\tilde{x} = (x^{\mathrm{T}}, 1)^{\mathrm{T}}$. Тогда на основе (3) в расширенном пространстве переменных можно записать следующую кусочно-линейную систему дифференциальных уравнений с автономными переключениями [22, с. 5–9]:

$$\dot{\tilde{x}} = \tilde{A}^{(i)}(t)\tilde{x} + \tilde{B}^{(i)}(t)u + \tilde{C}v^{(i)}, \quad \tilde{x} \in \Omega^{(i)} \times \{1\}, \ t \in [t_0, t_1], \tag{5}$$

$$\tilde{A}^{(i)}(t) = \begin{bmatrix} A^{(i)}(t) & f^{(i)}(t) \\ \mathbb{O}_{1\times n_x} & 0 \end{bmatrix}, \quad \tilde{B}^{(i)}(t) = \begin{bmatrix} B^{(i)}(t) \\ \mathbb{O}_{1\times n_u} \end{bmatrix}, \quad \tilde{C} = \begin{bmatrix} \mathbb{I}_{n_x \times n_x} \\ \mathbb{O}_{1\times n_x} \end{bmatrix},$$

где величина $v^{(i)}$ интерпретируется как помеха. Будем называть помеху допустимой, если она является измеримой функцией от времени и, кроме того, в каждый момент времени удовлетворяет ограничению $v^{(i)}(t) \in \mathcal{Q}^{(i)}(t)$. Индекс $i = i(x(t))$ в (5) является функцией состояния системы в момент времени t, однако для краткости записи аргументы этой функции будем опускать.

4. Функция цены

4.1. Общие сведения

Рассмотрим вспомогательную функцию цены:

$$\bar{V}(t,x) = \min_{u(\cdot)} \{\phi_{\mathcal{X}_1}(x(t_1)) : \ x(t) = x\}, \tag{6}$$

где $x(\cdot)$ – траектория нелинейной системы (1), выпущенная в прямом времени из начальной позиции $\{t, x\}$, $x \in \Omega$, при фиксированном позиционном управлении $u(\cdot)$. С помощью функции цены можно построить внутреннюю оценку множества разрешимости [15]:

$$\mathcal{W}(t, t_1, \mathcal{X}_1) = \left\{x \in \Omega : \ \bar{V}(t,x) \leqslant 0\right\}. \tag{7}$$

Наряду с (7) будем рассматривать оценку окрестности множества разрешимости:

$$\mathcal{W}_\varepsilon(t, t_1, \mathcal{X}_1) = \left\{x \in \Omega : \ \bar{V}(t,x) \leqslant \varepsilon\right\},$$
$$\mathcal{W}_\varepsilon(t, t_1, \mathcal{X}_1) = \left\{x \in \Omega \mid \exists u(\cdot) : \ \phi_{\mathcal{X}_1}(x(t_1; t, x)|_{u(\cdot)}) \leqslant \varepsilon\right\}.$$

Также известно, что в точке дифференцируемости (t, x), где $t < t_1$, $x \in \Omega$, функция $\bar{V}(t,x)$ удовлетворяет попятному уравнению Гамильтона–Якоби–Беллмана следующего вида:

$$\min_{u \in \mathcal{P}} \bar{V}'\left(t, x; \left(1, (\mathbf{f}(t,x) + \mathbf{g}(t,x)u)^{\mathrm{T}}\right)^{\mathrm{T}}\right) = 0, \tag{8}$$

где $\bar{V}'(t, x; \ell)$ – производная функции $\bar{V}(t,x)$ в точке (t,x) по направлению $\ell \in \mathbb{R}^{n_x+1}$. В конечный момент времени справедливо соотношение $\bar{V}(t_1, x) = \phi_{\mathcal{X}_1}(x)$. Функция $\bar{V}(t,x)$ может не быть непрерывно дифференцируемой, а решение уравнения (8) следует понимать в обобщенном смысле [23]. Однако можно заменить решение $\bar{V}(t,x)$ такой кусочно-квадратичной функцией, что уравнение (8) будет выполняться приближенно. Эта функция будет найдена далее на основе рассмотрения кусочно-линейной системы (5).

4.2. Кусочно-квадратичная функция

В каждой вершине $g_l^{(i)}$ каждого симплекса $\Omega^{(i)}$ определим аффинную по x функцию $\langle k_l^{(i)}(t), \tilde{x}\rangle$, где при любом фиксированном $t \in [t_0, t_1]$ вектор $k_l^{(i)} \in$

$\in \mathbb{R}^{n_x+1}$ – это вектор неизвестных параметров. Тогда для каждого симплекса $\Omega^{(i)}$ можно определить матрицу параметров, структура которой соответствует набору вершин $g_1^{(i)}, \ldots, g_{n_x+1}^{(i)}$:

$$K^{(i)}(t) = \left[k_1^{(i)}(t), \ldots, k_{n_x+1}^{(i)}(t)\right] \in \mathbb{R}^{(n_x+1)\times(n_x+1)}.$$

Определим кусочно-квадратичную функцию следующим образом:

$$V^{(i)}(t,\tilde{x}) = \langle \tilde{x}, K^{(i)}(t)\tilde{H}^{(i)}\tilde{x}\rangle, \quad \tilde{x} = (x^{\mathrm{T}}, 1)^{\mathrm{T}}, \quad x \in \Omega^{(i)}. \tag{9}$$

Формула (9) соответствует интерполяции рассмотренных аффинных функций в вершинах симплексов:

$$V^{(i)}(t,\tilde{x}) = \langle \tilde{x}, K^{(i)}(t)\tilde{H}^{(i)}\tilde{x}\rangle = \langle (K^{(i)}(t))^{\mathrm{T}}\tilde{x}, \alpha^{(i)}(x)\rangle = \sum_{l=1}^{n_x+1} \alpha_l^{(i)}(x)\langle k_l^{(i)}(t), \tilde{x}\rangle.$$

Отметим, поскольку функция (9) определена для расширенного пространства переменных $\tilde{x} = (x^{\mathrm{T}}, 1)^{\mathrm{T}}$, в таком виде может быть представлена произвольная кусочно-квадратичная функция, заданная на множестве симплексов $\Omega^{(i)}$, $i = \overline{1, N}$.

Будем использовать кусочно-аффинные управления вида

$$u(t,x) = Y^{(i)}(t)\tilde{H}^{(i)}\tilde{x} = \sum_{k=1}^{n_x+1} \alpha_k^{(i)}(x) y_k^{(i)}(t) \in \mathbb{R}^{n_u}, \tag{10}$$

где матрица $Y^{(i)}(t) \in \mathbb{R}^{n_u\times(n_x+1)}$ составлена из столбцов $y_k^{(i)}(t) \in \mathcal{P}$ – значений управлений в вершинах симплекса $\Omega^{(i)}$. Эти значения будут выбраны далее. При этом величины $y_k^{(i)}(t)$, соответствующие одной и той же вершине в различных симплексах, будут совпадать, т.е. управление $u(t,x)$ непрерывно по x. Заметим, что в силу выпуклости множества $\mathcal{P}$ достигается условие $u(t,x) \in \mathcal{P}$.

Запишем производную функции $V^{(i)}(t,\tilde{x})$ по направлению $\ell = (\ell_t, \ell_x) \in \in \mathbb{R}^{n_x+2}$:

$$\frac{dV^{(i)}}{d\ell} = \ell_t\langle \tilde{x}, \dot{K}^{(i)}\tilde{H}^{(i)}\tilde{x}\rangle + \langle \ell_x, [K^{(i)}\tilde{H}^{(i)} + (\tilde{H}^{(i)})^{\mathrm{T}}(K^{(i)})^{\mathrm{T}}]\tilde{x}\rangle. \tag{11}$$

В [13] было показано, что при $\ell = (\ell_t, \ell_x)^{\mathrm{T}}$, где $\ell_t = 1$, $\ell_x = \tilde{A}^{(i)}\tilde{x} + \tilde{B}^{(i)}u + \tilde{C}v^{(i)}$, справедлива оценка

$$\frac{dV^{(i)}}{d\ell}(t,\tilde{x}) \leqslant \langle \tilde{x}, [\dot{K}^{(i)} + Z^{(i)}]\tilde{H}^{(i)}\tilde{x}\rangle, \tag{12}$$

где матрица $Z^{(i)}$ известна и выражается через коэффициенты $K^{(i)}(t)$, коэффициенты $\tilde{A}^{(i)}(t)$, $\tilde{B}^{(i)}(t)$, $\tilde{C}$ кусочно-линейной системы (5), а также матрицы $Y^{(i)}(t)$, задающие управления в вершинах разбиения. Полученная оценка справедлива для любых допустимых помех $v^{(i)} \in \mathcal{Q}^{(i)}(t)$.

Приравнивая выражения $\dot{K}^{(i)} + Z^{(i)}$ к нулевой матрице, получим систему матричных дифференциальных уравнений, которая и задает изменение функции $V^{(i)}(t, \tilde{x})$ с течением времени:

$$\dot{K}^{(i)}(t) + Z^{(i)}(t) = 0, \quad t \in [t_0, t_1], \ i = \overline{1, N}. \tag{13}$$

Тогда из (12)–(13) следует, что вдоль любой траектории системы (5) в каждом симплексе $\Omega^{(i)}$ производная функции будет не возрастать. Далее будет показано, как модифицировать уравнения (13), чтобы полученная функция $V^{(i)}(t, \tilde{x})$ была непрерывной и, таким образом, невозрастание производной было бы обеспечено и при переходе через границу симплекса. Это может быть использовано для построения гарантированной априорной оценки отклонения конечной точки траектории от целевого множества.

4.3. Граничные условия

Для решения системы (13) необходимо задать граничные условия в конечный момент времени $t = t_1$. Для этого необходимо построить кусочно-квадратичную оценку сверху функции $\phi_{\mathcal{X}_1}$, представив которую в виде (9), можно определить $K^{(i)}(t_1)$. В частности, если границей множества $\mathcal{X}_1$ является гиперповерхность второго порядка, то справедливо представление $\phi_{\mathcal{X}_1}(x) = \langle \tilde{x}, \hat{K}\tilde{x} \rangle$ при некоторой матрице $\hat{K} = \hat{K}^{\mathrm{T}}$. Следовательно, в конечный момент времени в каждом симплексе можно выбрать значения параметров функции $V^{(i)}(t_1, \tilde{x})$, равные

$$K^{(i)}(t_1) = \hat{K}(\tilde{H}^{(i)})^{-1}. \tag{14}$$

В общем случае для любой дважды дифференцируемой функции $\phi_{\mathcal{X}_1}$ можно сконструировать кусочно-аффинную оценку сверху [12], которая является частным случаем кусочно-квадратичной и приводит к условиям типа (14). При этом функция $V^{(i)}(t, \tilde{x})$ в конечный момент времени $t = t_1$ будет непрерывной по $\tilde{x}$ на всем множестве $\Omega \times \{1\}$.

4.4. Сглаживание функции

Отметим, что при решении задачи Коши (13)–(14) функция $V^{(i)}(t, x)$, определяемая выражением (9), будет иметь разрывы на границах симплексов. Это связано с тем, что каждый столбец матрицы $K^{(i)}(t)$ определяет коэффициенты кусочно-аффинной функции $\langle k_l^{(i)}(t), \tilde{x} \rangle$ в некоторой вершине разбиения g_l, но каждая такая точка, вообще говоря, является вершиной сразу нескольких симплексов. Поскольку матрицы $Z^{(i)}$ в оценке (11) для каждого симплекса строятся независимо друг от друга, то значения производных $\dot{k}_l^{(i)}(t)$ определяются сразу несколькими несовместными условиями.

Таким образом, требуется модифицировать оценку (11), чтобы полученная функция $V^{(i)}(t, x)$ была непрерывна. Предложим альтернативный способ оценки матриц $Z^{(i)}$, нежели в [13].

Представим (13) в векторной форме, т.е. запишем дифференциальное уравнение для каждого столбца матрицы $K^{(i)}$:

$$\dot{k}_l^{(i)}(t) + z_l^{(i)}(t) = 0, \quad t \in [t_0, t_1], \quad i = \overline{1, N}, \quad l = \overline{1, n_x + 1}, \tag{15}$$

где $z_l^{(i)}$ – соответствующий столбец матрицы $Z^{(i)}$. Это позволяет переписать оценку (12) в следующем виде:

$$\begin{aligned} &\frac{dV^{(i)}}{d\ell}(t, \tilde{x}) \leqslant \langle \tilde{x}, [\dot{K}^{(i)} + Z^{(i)}]\tilde{H}^{(i)}\tilde{x}\rangle = \langle \tilde{x}, \dot{K}^{(i)}\tilde{H}^{(i)}\tilde{x}\rangle + \langle \tilde{x}, Z^{(i)}\tilde{H}^{(i)}\tilde{x}\rangle \leqslant \\ &\leqslant \langle \tilde{x}, \dot{K}^{(i)}\tilde{H}^{(i)}\tilde{x}\rangle + \langle \tilde{x}, Z^{(i)}\alpha^{(i)}(x)\rangle = \langle \tilde{x}, \dot{K}^{(i)}\tilde{H}^{(i)}\tilde{x}\rangle + \sum_{l=1}^{n_x+1} \alpha_l^{(i)}(x)\langle \tilde{x}, z_l^{(i)}\rangle. \end{aligned} \tag{16}$$

При каждом фиксированном $t \in [t_0, t_1]$ определим в каждой вершине $g_l^{(i)}$ следующую вспомогательную задачу линейного программирования относительно нового неизвестного вектора $\hat{z}_l^{(i)}$:

$$\begin{cases} \langle \hat{z}_l^{(i)}, \tilde{g}_l^{(i)}\rangle \to \min, \\ \langle \hat{z}_l^{(i)}, \tilde{g}_k^{(j)}\rangle \geqslant \langle z_{\nu(i,l,j)}^{(j)}, \tilde{g}_k^{(j)}\rangle \quad \forall j: \ g_l^{(i)} \in \Omega^{(i)} \cap \Omega^{(j)}, \ k = \overline{1, n_x + 1}, \end{cases} \tag{17}$$

где $\nu(i, l, j)$ – локальный номер вершины $g_l^{(i)} \in \Omega^{(i)} \cap \Omega^{(j)}$ в симплексе $\Omega^{(j)}$.

Из решений $\hat{z}_l^{(i)}$ аналогичным образом составим матрицы $\hat{Z}^{(i)}$. Учитывая ограничения задачи (17) и линейность рассматриваемых функций, можем продолжить неравенство (16):

$$\begin{aligned} &\frac{dV^{(i)}}{d\ell}(t, \tilde{x}) \leqslant \langle \tilde{x}, [\dot{K}^{(i)} + Z^{(i)}]\tilde{H}^{(i)}\tilde{x}\rangle \leqslant \langle \tilde{x}, \dot{K}^{(i)}\tilde{H}^{(i)}\tilde{x}\rangle + \sum_{l=1}^{n_x+1} \alpha_l^{(i)}(x)\langle \tilde{x}, z_l^{(i)}\rangle \leqslant \\ &\leqslant \langle \tilde{x}, \dot{K}^{(i)}\tilde{H}^{(i)}\tilde{x}\rangle + \sum_{l=1}^{n_x+1} \alpha_l^{(i)}(x)\langle \tilde{x}, \hat{z}_l^{(i)}\rangle = \langle \tilde{x}, [\dot{K}^{(i)} + \hat{Z}^{(i)}]\tilde{H}^{(i)}\tilde{x}\rangle. \end{aligned}$$

Заметим, что решения задач (17), соответствующие одной и той же вершине в различных симплексах $\Omega^{(i)}$, будут совпадать (в случае, если задача линейного программирования допускает неединственное решение, их можно выбрать одинаковыми). Следовательно, кусочно-заданная функция цены (9), полученная при решении задачи Коши

$$\begin{cases} \dot{K}^{(i)} + \hat{Z}^{(i)} = 0, & i = \overline{1, N}, \quad t \in [t_0, t_1], \\ K^{(i)}(t_1) = \hat{K}(\tilde{H}^{(i)})^{-1}, & i = \overline{1, N}, \end{cases} \tag{18}$$

будет непрерывной по $(t, \tilde{x})$ во всей рассматриваемой области. При этом функционал в задаче (17) соответствует значениям $V^{(i)}(t, \tilde{x})$ в вершинах симплексов и, таким образом, способствует уменьшению значений функции в этих точках.

5. Алгоритм выбора управления

Прежде чем приступить к решению задачи (18), необходимо определить управления $y_k^{(i)}$ из (10) в вершинах симплексов, чтобы на основе этих значений построить матрицы $\hat{Z}^{(i)}$. В [11–13] они выбирались так, чтобы в каждом симплексе $\Omega^{(i)}$ минимизировать производную (11) функции $V^{(i)}(t, \tilde{x})$ вдоль траектории движения, но с учетом кусочно-заданного характера этой функции возникала неоднозначность при выборе значений $y_k^{(i)}$. Для ее устранения приходилось дополнительно корректировать управления, что негативно сказывалось на полученном решении.

В данной работе на примере обучения с подкреплением демонстрируется, что метод допускает использование управлений, полученных на основе альтернативных подходов, в результате чего построенная аппроксимация функции цены (6) может оказаться более точной.

Обучение с подкреплением [16] – это раздел машинного обучения, в котором поведение агента корректируется при многократном взаимодействии с окружающей средой в зависимости от получаемых от нее вознаграждений при каждом совершенном действии. Применительно к рассматриваемой задаче агент реализует управляющую стратегию $u = u(t, x)$, а в качестве функции мгновенного вознаграждения выберем

$$\mathcal{L}(t, x) = \begin{cases} 0, & t < t_1, \\ -d^2(x, \mathcal{X}_1), & t = t_1, \end{cases} \tag{19}$$

где $d(x, \mathcal{X}_1)$ обозначает расстояние от точки x до множества $\mathcal{X}_1$.

Proximal Policy Optimization (PPO) – это один из методов обучения с подкреплением, где стратегия управления представлена с помощью нейронной сети, веса которой обновляются методом градиентного спуска при оптимизации некоторого функционала качества. Функционал качества основан на максимизации кумулятивного вознаграждения по окончании эксперимента, однако представляет собой более сложное выражение [17], чтобы обеспечить стабильный процесс обучения.

Преимуществом алгоритма PPO является возможность его применения к непрерывным системам, в том числе к системам вида (1). Этим свойством обладают и некоторые другие алгоритмы, например DDPG [24] и SAC [25]. Они также могут быть использованы в предложенном подходе, однако в рассмотренных далее примерах показали меньшую точность при переводе системы (1) в окрестность целевого множества.

Пусть множество $\mathcal{P}$ допускает конечномерную параметризацию, в таком случае вектор $u \in \mathcal{P}$ определяется набором параметров $\theta \in \mathbb{R}^r$, где $\theta_i \in \in [\theta_i^{\min}, \theta_i^{\max}]$, $i = \overline{1, r}$, и цель заключается в определении этого набора для каждой фиксированной позиции (t, x). Но поскольку алгоритм PPO рассчитан на стохастические стратегии, обычно предполагается, что θ – это случайный вектор, имеющий многомерное нормальное распределение $\theta \sim \mathcal{N}(\mu, \Sigma)$

с диагональной матрицей ковариации. При использовании алгоритма сперва обучается нейронная сеть, которая предсказывает параметры этого распределения, а затем, во время расчета значений $u(t,x)$, генерируются реализации соответствующего случайного вектора. Имея обученную нейронную сеть, легко получить детерминированное управление: для этого вместо генерации случайного вектора достаточно взять соответствующие математические ожидания: $\theta = \mu$.

Отметим, что на значения параметров θ_i наложены интервальные ограничения, в то время как носителем нормального случайного вектора является все пространство $\mathbb{R}^r$. Чтобы удовлетворять требованиям, на практике значения параметров "обрезаются" [26] и новые значения получаются по формуле $\tilde{\theta}_i = \min\{\theta_i^{\max}, \max\{\theta_i, \theta_i^{\min}\}\}$, хотя допускается использование других преобразований. Кроме того, для указанных случайных величин можно использовать распределения с ограниченным носителем [27].

Такие детерминированные управления на основе нейросетевой модели, удовлетворяющие ограничению (2), будем обозначать как $\hat{u}(t,x)$. Результирующее кусочно-аффинное управление, используемое в данной работе, определяется по формуле (10):

$$u(t,x) = \sum_{k=1}^{n_x+1} \alpha_k^{(i)}(x)\hat{u}(t, g_k^{(i)}), \quad x \in \Omega^{(i)}. \tag{20}$$

Заметим, что в силу устройства нейросети функция $\hat{u}(t,x)$ будет непрерывной по (t,x). Отсюда следует, что при стремлении диаметра разбиения множества Ω на симплексы к нулю итоговое управление (20) будет поточечно сходиться к $\hat{u}(t,x)$.

6. Основной результат

Введение вышеописанных конструкций позволяет доказать следующую теорему.

Теорема 1. Пусть матричные функции $K^{(i)}(t) \in \mathbb{R}^{(n_x+1)\times(n_x+1)}$ являются решением задачи Коши (18). *Пусть $V(t,\tilde{x})$ – непрерывная кусочно-квадратичная функция, определенная на множестве $[t_0,t_1]\times\Omega\times\{1\}$, которая в каждом симплексе $\Omega^{(i)}$ задается равенством $V^{(i)}(t,\tilde{x}) = \langle \tilde{x}, K^{(i)}(t)\tilde{H}^{(i)}\tilde{x}\rangle$. Тогда множество $\mathcal{W}_\varepsilon^{int}(t_0) = \left\{ x \in \Omega \;\middle|\; V(t_0,\tilde{x}) \leqslant \varepsilon \right\}$ (в предположении его непустоты) является внутренней оценкой множества разрешимости исходной нелинейной системы* (1):

$$\mathcal{W}_\varepsilon^{int}(t_0) \subseteq \mathcal{W}_\varepsilon(t_0,t_1,\mathcal{X}_1).$$

Доказательство теоремы основано на анализе траекторий нелинейной системы (1), замкнутой управлением вида (10), однако не зависит от способа нахождения векторов $y_k^{(i)}(t) \in \mathcal{P}$ в вершинах симплексов и проходит по схеме, приведенной в [13].

7. Примеры работы алгоритма

7.1. Нелинейная система

Рассмотрим движение маятника на тележке с учетом силы трения [28], которое описывается системой уравнений

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = -w^2 \sin(x_1) - 2\gamma x_2 - w^2 \cos(x_1) u, \end{cases} \tag{21}$$

где ω и γ являются параметрами, x_1 и x_2 – угол отклонения маятника и угловая скорость соответственно, управление u соответствует ускорению тележки. Пусть $\omega = 1$, $\gamma = 0{,}1$ и требуется перевести систему из начального положения $(-0{,}3,\ 0{,}6)^{\mathrm{T}}$ при $t_0 = 0$ в малую окрестность начала координат в момент времени $t_1 = 1$. На управление наложено ограничение $u \in [-1, 1]$.

В качестве простейшей модели нейронных сетей, используемых в алгоритме PPO, предлагается выбрать двухслойный перцептрон [29] с функцией активации $\tanh(x)$. При обучении было сгенерировано 10 000 пробных траекторий системы (21), стартующих из различных случайных точек $x^0 \in \Omega$ в момент времени t_0, и стратегия управления $\hat{u}(t, x)$ обновлялась на основе штрафов (19). На рис. 1 представлена траектория, полученная при использовании алгоритма PPO без дополнительных модификаций. Расстояние между конечной точкой траектории и началом координат составляет 0,027.

Для расчета кусочно-квадратичной функции (9) сперва были зафиксированы вершины $g_k \in \mathbb{R}^2$, расположенные на прямоугольной сетке со сторонами длины $\Delta = 0{,}1$, которые затем были использованы для разбиения множества $\Omega = [-1, 1] \times [-1, 1]$ на $N = 800$ равных симплексов. На рис. 2 представле-

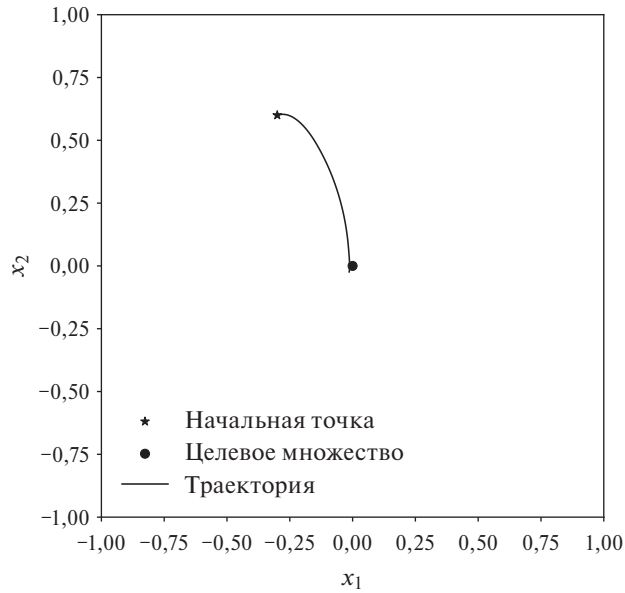

Рис. 1. Траектория на основе нейросетевого управления $\hat{u}(t, x)$.

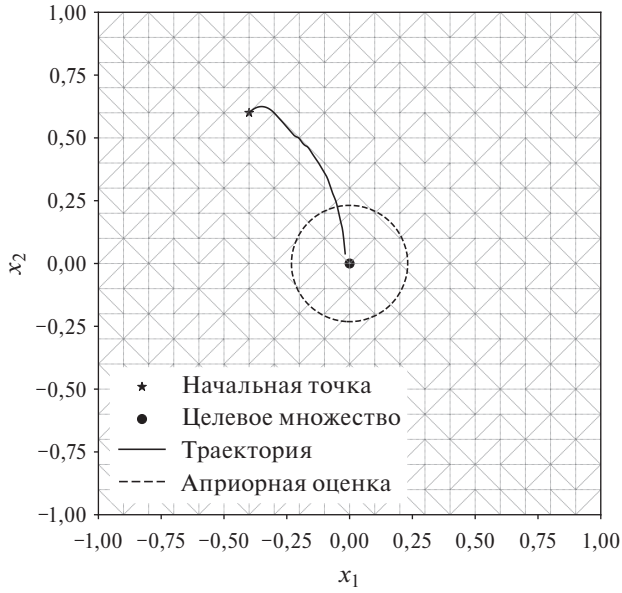

Рис. 2. Траектория на основе управления из [13].

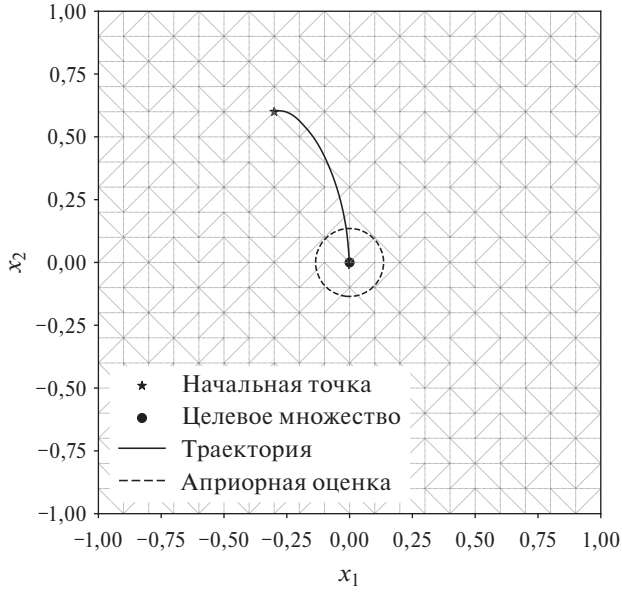

Рис. 3. Траектория на основе аппроксимации (20) нейросетевого управления $\hat{u}(t, x)$.

ны результаты, полученные с помощью алгоритма выбора управлений (10) из [13]: пунктирная линия обозначает границу множества, куда априорно гарантируется попадание траектории системы; расстояние между $x(t_1)$ и целевым множеством составляет 0,043.

На рис. 3 представлена траектория, полученная описанной в текущей работе комбинацией методов при том же разбиении на симплексы. Расстояние

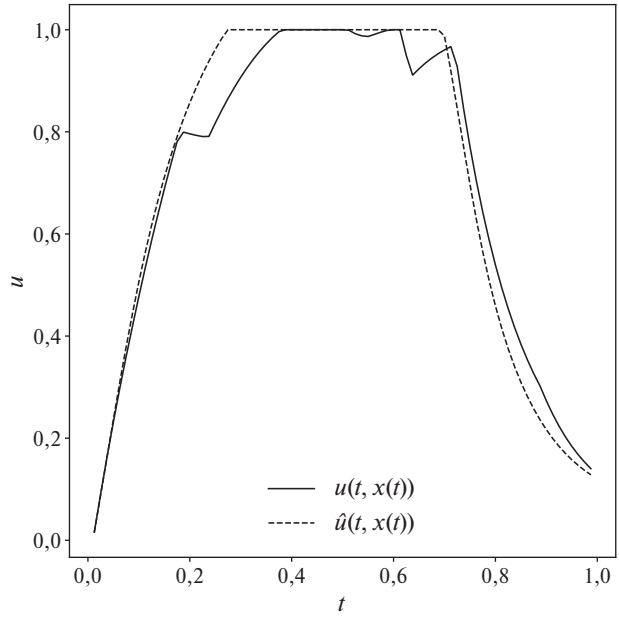

Рис. 4. Нейросетевое управление $\hat{u}(t, x(t))$ и результирующее управление $u(t, x(t))$.

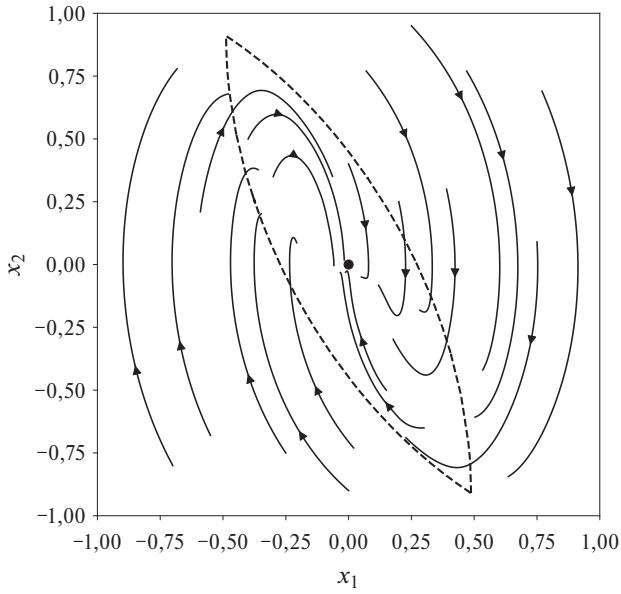

Рис. 5. Граница множества разрешимости при $t = t_0$ и траектории системы (21) при замыкании результирующим управлением $u(t, x)$.

до начала координат в этом случае равно 0,023, при этом изменение ошибки объясняется разницей между исходным нейросетевым управлением $\hat{u}(t, x)$ и его аппроксимацией (20). На рис. 4 приведены управления, соответствующие траекториям, изображенным на рис. 1 и рис. 3. Видно, что априорная

погрешность представленного метода меньше, чем в алгоритме [13]. Приведенный пример подтверждает, что в каждом из случаев априорная оценка, полученная из функции цены (9), является гарантированной.

На рис. 5 непрерывными линиями обозначены траектории, полученные предложенным методом при старте из различных начальных точек; стрелками обозначено направление движения вдоль траекторий. Кроме того, пунктирной линией обозначена граница множества разрешимости в классе кусочно-непрерывных программных управлений, вычисленная на основе принципа максимума Л.С. Понтрягина [30, с. 336–344].

7.2. Линейная система

Чтобы получить более полное представление о точности предложенного подхода, рассмотрим линейную систему

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = u \tag{22}$$

на отрезке $t \in [0, 1]$. В данном случае не требуется применять описанный ранее механизм кусочной линеаризации, однако такая система хорошо изучена в литературе (например, в [30]). Пусть управление удовлетворяет ограничению $u \in [-2, 2]$ и требуется перевести систему в начало координат в момент времени $t = 1$. Тогда может быть получено, что точка $x^0 = (0{,}5,\ 0)^{\mathrm{T}}$ лежит на границе множества разрешимости в момент $t = 0$ и достигается на кусочно-постоянном управлении $u^*(t) = 2\,\mathrm{sign}(t - 0{,}5)$.

Для численных экспериментов была выбрана нейросетевая модель той же структуры, что и в предыдущем примере. Модель обучалась на персональном компьютере в течение одного часа, после чего при зафиксированных весах нейросети при различных диаметрах разбиения на симплексы $\Omega^{(i)}$ были построены кусочно-квадратичные функции вида (9). Рассматривалось множество $\Omega = [-1{,}5,\ 1{,}5] \times [-1{,}5,\ 1{,}5]$.

На рис. 6 обозначены априорная оценка попадания в начало координат из точки x^0 при шаге прямоугольной сетки $\Delta = 0{,}25$ (что соответствует разбиению на 288 представленных на рисунке симплексов) и полученная траектория, а на рис. 7 приведено соответствующее управление $u(t, x(t))$ вида (20). На рис. 8 изображено множество разрешимости, вычисленное на основе принципа максимума Л.С. Понтрягина, а также траектории, полученные предложенным методом при старте из различных начальных точек.

На рис. 9 для той же начальной точки $x^0 = (0{,}5,\ 0)^{\mathrm{T}}$ приведены зависимости априорной и апостериорной погрешностей от числа симплексов разбиения $\Omega^{(i)}$ множества Ω. При уменьшении диаметра сетки апостериорное значение погрешности сходится к 0,104, что соответствует точности при исходном нейросетевом управлении $\hat{u}(t, x)$. Отметим, что эта точность может быть повышена за счет рассмотрения других нейросетевых моделей, возможно, с бóльшим числом параметров. Кроме того, из рис. 9 следует, что априорная погрешность снижается, однако с некоторого момента вновь начинает возрастать. Такое увеличение погрешности объясняется несовершенством вспомо-

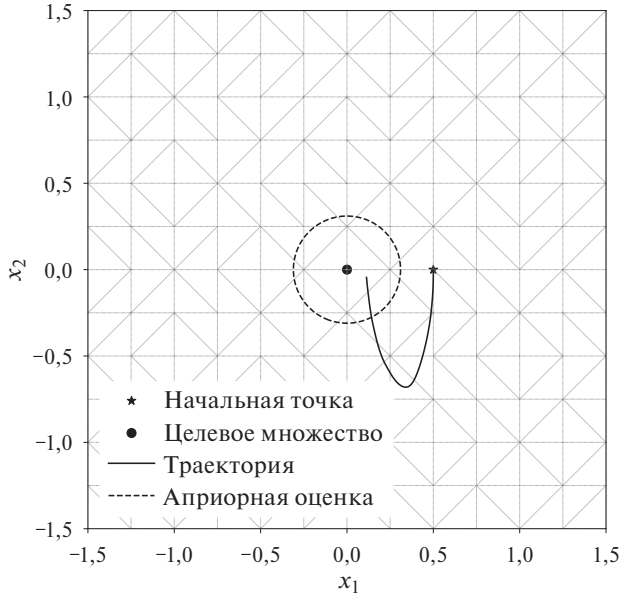

Рис. 6. Траектория системы (22) и априорная оценка попадания в целевую точку.

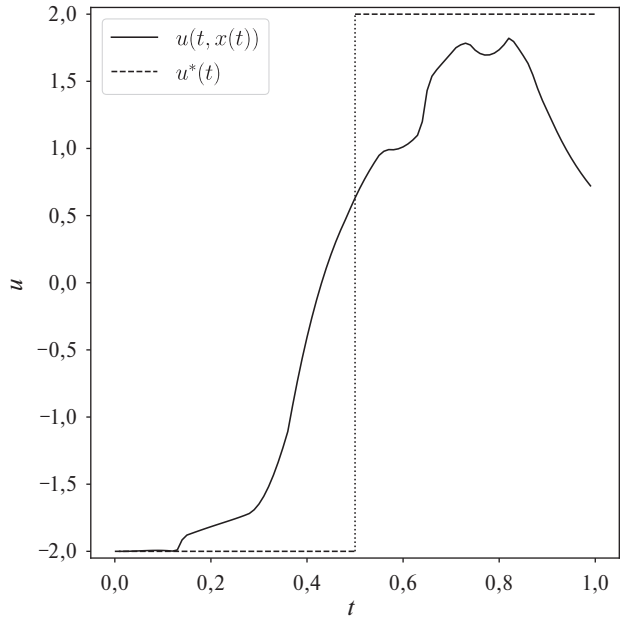

Рис. 7. Результирующее управление $u(t, x(t))$ для системы (22), а также оптимальное управление $u^*(t)$.

гательных задач оптимизации (17): их решения в соседних вершинах могут значительно отличаться друг от друга, что влияет на устойчивость метода при мелком диаметре разбиения. Эта проблема может быть устранена за

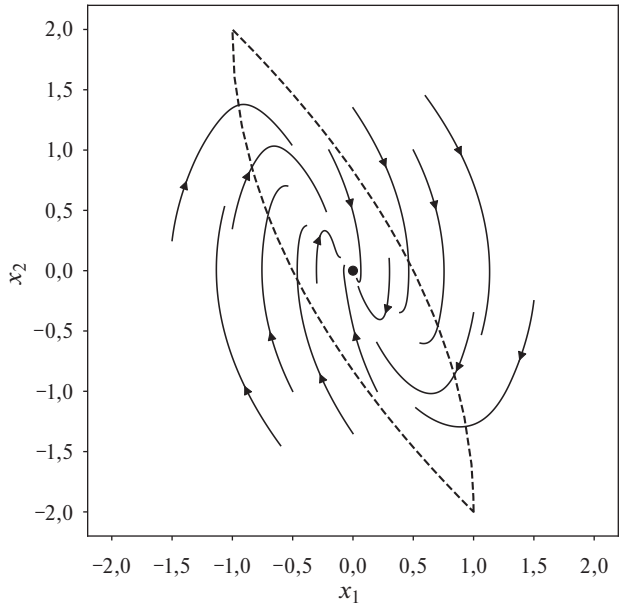

Рис. 8. Граница множества разрешимости при $t = t_0$ и траектории системы (22) при замыкании результирующим управлением $u(t, x)$.

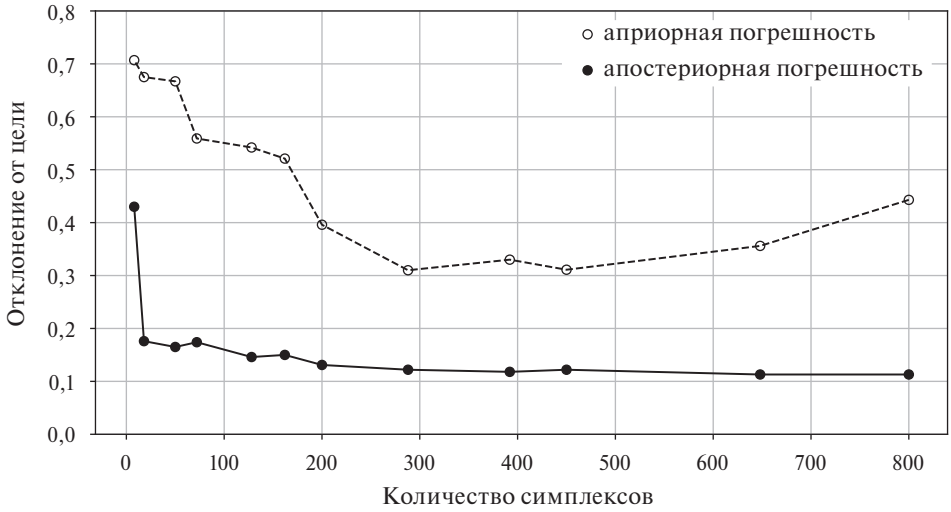

Рис. 9. Отклонение от целевой точки $x^1 = (0, 0)^{\mathrm{T}}$ в зависимости от количества симплексов разбиения.

счет замены функционала в (17) или же за счет введения дополнительных "регуляризующих" слагаемых в систему (18), использование которых было предложено в [11].

На рис. 10 указано время вычисления функции цены в зависимости от числа симплексов при фиксированной нейросетевой стратегии $\hat{u}(t, x)$. Можно заметить, что временные затраты линейно растут с увеличением количества

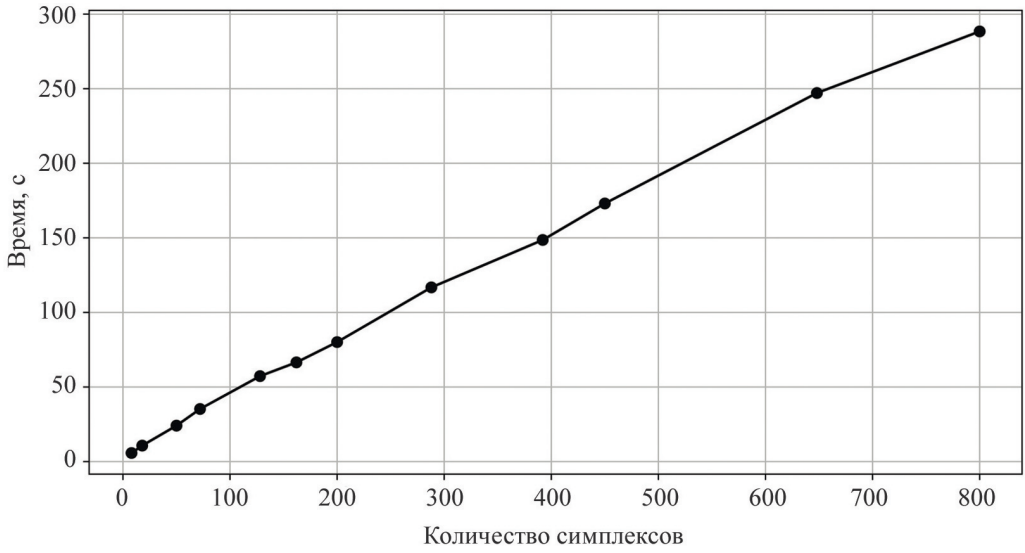

Рис. 10. Время вычисления функции цены при фиксированной функции $\hat{u}(t, x)$.

симплексов, и при не слишком мелком диаметре разбиения время вычислений мало в сравнении со временем обучения нейронной сети.

8. Заключение

Приведенные в данной работе формулы позволяют получить позиционную стратегию управления, решающую задачу приближенно, и кусочно-аффинную аппроксимацию этой стратегии на множестве симплексов. Последняя используется для построения непрерывной кусочно-квадратичной функции, задающей внутреннюю оценку множества разрешимости в задаче целевого управления. Для полученного кусочно-аффинного управления справедлива гарантированная априорная оценка погрешности попадания траектории в целевое множество. Предложенный подход может быть использован при решении задач управления нелинейными системами с небольшой размерностью фазового пространства.

СПИСОК ЛИТЕРАТУРЫ

1. *Незнахин А.А., Ушаков В.Н.* Сеточный метод приближенного построения ядра выживаемости для дифференциального включения // Журн. вычисл. мат. и мат. физики. 2001. Т. 41. № 6. С. 895–908.
2. *Goubault E., Putot S.* Inner and Outer Reachability for the Verification of Control Systems // Proceedings of the 22nd ACM International Conference on Hybrid Systems: Computation and Control. 2019. P. 11–22. https://doi.org/10.1145/3302504.3311794
3. *Shafa T., Ornik M.* Reachability of Nonlinear Systems with Unknown Dynamics. 2021. https://doi.org/10.48550/arXiv.2108.11045

4. *Garrido S., Moreno L.E., Blanco D., Jurewicz P.P.* Optimal control using the Fast Marching Method // 35th Annual Conference of IEEE Industrial Electronics. 2009. P. 1669–1674. https://doi.org/10.1109/IECON.2009.5414750

5. *Субботина Н.Н., Токманцев Т.Б.* Классические характеристики уравнения Беллмана в конструкциях сеточного оптимального синтеза // Тр. мат. ин-та им. В.А. Стеклова. 2010. Т. 271. С. 259–277.

6. *Xue B., Fränzle M., Zhan N.* Inner-Approximating Reachable Sets for Polynomial Systems with Time-Varying Uncertainties // IEEE Transact. Autom. Control. 2019. V. 65. No. 4. P. 1468–1483. https://doi.org/10.1109/TAC.2019.2923049.

7. Lee D., Tomlin C.J. Efficient Computation of State-Constrained Reachability Problems Using Hopf–Lax Formulae // IEEE Transact. Autom. Control. 2023. P. 1–15. https://doi.org/10.1109/TAC.2023.3241180

8. *Cheng T., Lewis F.L., Abu–Khalaf M.* Fixed-Final-Time-Constrained Optimal Control of Nonlinear Systems Using Neural Network HJB Approach // IEEE Transactions on Neural Networks. 2007. V. 18. No. 6. P. 1725–1737. https://doi.org/10.1109/TNN.2007.905848

9. *Onken D., Nurbekyan L., Li X., et al.* A Neural Network Approach for High-Dimensional Optimal Control Applied to Multiagent Path Finding // IEEE Transact. Control Syst. Techn. 2023. V. 31. No. 1. P. 235–251. https://doi.org/10.1109/TCST.2022.3172872

10. Sánchez–Sánchez C., Izzo D., Hennes D. Learning the optimal state-feedback using deep networks // 2016 IEEE Symposium Series on Computational Intelligence. 2016. P. 1–8. https://doi.org/10.1109/SSCI.2016.7850105

11. *Tochilin P.A.* Piecewise affine feedback control for approximate solution of the target control problem // IFAC-PapersOnLine. 2020. V. 53. No. 2. P. 6127–6132. https://doi.org/10.1016/j.ifacol.2020.12.1691

12. *Точилин П.А.* О построении кусочно-аффинной функции цены в задаче оптимального управления на бесконечном отрезке времени // Тр. ин-та мат. и механики УрО РАН. 2020. Т. 26. № 1. С. 223–238. https://doi.org/10.21538/0134-4889-2020-26-1-223-238

13. *Чистяков И.А., Точилин П.А.* Применение кусочно-квадратичных функций цены для приближенного решения нелинейной задачи целевого управления // Дифференциальные уравнения. 2020. Т. 56. № 11. С. 1545–1554. https://doi.org/10.1134/S0374064120110126

14. *Куржанский А.Б.* Принцип сравнения для уравнений типа Гамильтона–Якоби в теории управления // Тр. ин-та мат. и механики УрО РАН. 2006. Т. 12. № 1. С. 173–183.

15. *Kurzhanski A.B., Varaiya P.* Dynamics and control of trajectory tubes. Theory and computation. Birkhäuser, 2014. https://doi.org/10.1007/978-3-319-10277-1

16. *Саттон Р.С., Барто Э.Г.* Обучение с подкреплением. М.: ДМК пресс, 2020.

17. *Schulman J., Wolski F., Dhariwal P., et al.* Proximal policy optimization algorithms. 2017. https://doi.org/10.48550/arXiv.1707.06347

18. *Пшеничный Б.Н.* Выпуклый анализ и экстремальные задачи. М.: Наука, 1980.

19. *Скворцов А.В., Мирза Н.С.* Алгоритмы построения и анализа триангуляции. Томск: Изд-во Том. ун-та, 2006.

20. *Rajan V.T.* Optimality of the Delaunay triangulation in $\mathbb{R}^d$ // Discrete & Computational Geometry. 1994. V. 12. No. 2. P. 189–202. https://doi.org/10.1007/BF02574375
21. *Brown K.Q.* Voronoi diagrams from convex hulls // Inform. Proc. Lett. 1979. V. 9. No. 5. P. 223–228. https://doi.org/10.1016/0020-0190(79)90074-7
22. *Liberzon D.* Switching in Systems and Control. Birkhauser, 2003. https://doi.org/10.1007/978-1-4612-0017-8
23. *Bardi M., Capuzzo-Dolcetta I.* Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations. Ser. Systems & Control: Foundations & Applications. Boston: Birkhäuser, 2008. https://doi.org/10.1007/978-0-8176-4755-1
24. *Lillicrap T.P., Hunt J.J., Pritzel A., et al.* Continuous control with deep reinforcement learning. 2019. https://doi.org/10.48550/arXiv.1509.02971
25. *Haarnoja T., Zhou A., Abbeel P., Levine S.* Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. 2018. https://doi.org/10.48550/arXiv.1801.01290
26. *Raffin A., Hill A., Gleave A., et al.*, Stable-Baselines3: Reliable Reinforcement Learning Implementations // J. Machin. Lear. Res. 2021. V. 22. No. 268. P. 1–8.
27. *Petrazzini I.G.B., Antonelo E.A.* Proximal Policy Optimization with Continuous Bounded Action Space via the Beta Distribution // 2021 IEEE Symposium Series on Computational Intelligence (SSCI). 2022. P. 1–8. https://doi.org/10.1109/SSCI50451.2021.9660123
28. *Reissig G.* Computing Abstractions of Nonlinear Systems // IEEE Transact. Autom. Control. 2011. V. 56. No. 11. P. 2583–2598. https://doi.org/10.1109/TAC.2011.2118950
29. *Голубев Ю.Ф.* Нейронные сети в мехатронике // Фундамент. и прикл. матем. 2005. Т. 11. № 8. С. 81–103.
30. *Ли Э.Б., Маркус Л.* Основы теории оптимального управления. М.: Наука, 1972.

Статья представлена к публикации членом редколлегии П.В. Пакшиным.

Поступила в редакцию 29.08.2023

После доработки 14.10.2024

Принята к публикации 29.10.2024