ЖУРНАЛ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И МАТЕМАТИЧЕСКОЙ ФИЗИКИ, 2021, том 61, № 5, с. 865–877

ОПТИМАЛЬНОЕ УПРАВЛЕНИЕ

УДК 517.977.54

TT-QI: УСКОРЕННАЯ ИТЕРАЦИЯ ФУНКЦИИ ЦЕННОСТИ В ФОРМАТЕ ТЕНЗОРНОГО ПОЕЗДА ДЛЯ ЗАДАЧ СТОХАСТИЧЕСКОГО ОПТИМАЛЬНОГО УПРАВЛЕНИЯ[1)]

© 2021 г. А. И. Бойко[1,*], И. В. Оселедец[1,2,], Г. Феррер[1]**

[1] *121205 Москва, Большой бульвар, 30, стр. 1, Сколковский институт науки и технологий, Россия*

[2] *119333 Москва, ул. Губкина, 8, ИВМ РАН, Россия*

**e-mail: alexey.boyko@skolkovotech.ru*

***e-mail: i.oseledets@skoltech.ru*

Поступила в редакцию 24.11.2020 г.
Переработанный вариант 24.11.2020 г.
Принята к публикации 14.01.2021 г.

Рассматривается задача стохастического оптимального управления общего вида с малым винеровским шумом. Данная задача аппроксимируется с помощью марковского процесса принятия решений. Решение уравнения Беллмана на функцию ценности вычисляется с помощью метода итерации ценности (VI) в формате малорангового тензорного поезда (TT-VI). Предложена модификация данного алгоритма (TT-QI): нелинейный оператор Беллмана итеративно применяется сначала с использованием малоранговых алгебраических операций, а затем с использованием алгоритма крестовой аппроксимации. Показана более низкая, чем в основном методе, сложность на одну итерацию в случае малых TT-рангов тензоров вероятностей перехода. На примере задач управления обратным маятником и машинами Дубинса показано ускорение времени расчета оптимального регулятора в 3–10 раз по сравнению с существующим методом. Библ. 13. Фиг. 6. Табл. 1.

Ключевые слова: динамическое программирование, оптимальное управление, марковские процессы принятия решений, малоранговые разложения.

DOI: 10.31857/S0044466921050045

1. ВВЕДЕНИЕ

Задачи оптимального управления часто возникают в различных областях робототехники. Для многих типовых задач были разработаны решения с помощью оптимизации PID-регуляторов, линейно-квадратичных регуляторов, либо более общих уравнения Риккатти и принципа максимума Понтрягина.

Однако для многих новых задач робототехники, таких как динамическое управление шагающими роботами на ландшафте произвольной формы или акробатическое маневрирование колесными и летающими роботами далеко за областью линейности и при воздействии случайных возмущений, синтез оптимального регулятора в общем виде может быть осуществлен только с помощью уравнения Беллмана.

Стохастические динамические системы могут быть представлены (с точностью до погрешности дискретизации) как марковский процесс принятия решений (см. [1], [2]). Формулировка задачи оптимального управления на языке уравнения Беллмана также позволяет нахождение оптимального регулятора в случае произвольной нелинейной, разрывной или точечной награды. Это делает возможным переписать на беллмановский язык такие задачи, как терминальную задачу об оптимальном быстродействии, задачу о минимальном затраченном топливе (с произвольной, а не только квадратичной характеристикой), задачу максимизации вероятности попадания в целевую область.

Ключевой проблемой в таком подходе является тот факт, что решение задачи стохастического оптимального управления на языке Беллмана страдает от “проклятия размерности”, так как

[1)]Работа выполнена при частичной финансовой поддержке Минобрнауки РФ (проект 14.756.31.0001).

имеет крайне высокую асимптотику вычислительной сложности. Сложность по памяти растет как $O(N^{d_s+d_a})$, где N – количество узлов дискретизации сетки по одной координате, а d_s и d_a – размерности пространства состояний и пространства управляющих сигналов соответственно. Например, для простого беспилотного летательного аппарата $d_s + d_a = 16$.

Один из методов обойти данную проблему – использовать дифференцируемые функциональные аппроксиматоры, например нейронные сети. Такой подход изначально был развит Бертсекасом под названием *нейродинамического программирования* (см. [3]), а позже стал известен как (глубокое) машинное обучение с подкреплением. К настоящему моменту действие в этом направлении привело к значительным успехам в решении задач управления существенно нелинейными малоприводными системами, в том числе с неголономными связями (см. [4]). Пожалуй, самым выдающимся примером этого является синтез регулятора для движения бега в подробной многозвенной малоприводной математической модели человека, управляемой более чем 100 мышцами (см. [5]). Такой подход, однако, имеет свои недостатки. Главным образом, это очень высокие требования к количеству данных (симуляций Монте-Карло) и к вычислительным мощностям, а также необходимость вручную подбирать различные параметры оптимизатора и топологию нейросети и отсутствие гарантий сходимости оптимального управления к глобальному оптимуму даже после длительной оптимизации.

Другой способ обойти "проклятие размерности" уравнения Беллмана – это использовать малоранговые тензорные разложения. Эффективность такого подхода для решения задачи оптимального управления была продемонстрирована Н. Хоровицем (см. [6]) из Калифорнийского технологического института с использованием линеаризованного уравнения Гамильтона–Якоби–Беллмана для контрольно-аффинных систем совместно с каноническим тензорным разложением. Несколько иной метод был предложен А. Городетским из Массачуссетского технологического института с использованием нелинейного уравнения Беллмана для систем общего вида и TT-разложения (см. [7]) и его дифференцируемого обобщения (см. [8], [9]). В этих статьях были найдены решения различных нелинейных малоприводных задач управления единообразным подходом, в том числе задачи акробатического пилотажа квадрокоптером далеко за областью линейности, используя лишь однопроцессорную многоядерную рабочую станцию. В данной статье мы рассматриваем именно подход, описанный А. Городетским.

2. ЗАДАЧА СТОХАСТИЧЕСКОГО ОПТИМАЛЬНОГО УПРАВЛЕНИЯ

Рассмотрим стохастическую динамическую полностью наблюдаемую систему, задаваемую уравнением

$$ds_i(t) = b_i(s,a)dt + \sigma_{ij}(s)dw. \tag{1}$$

Система в каждый момент времени описывается вектором состояния $s \in \mathcal{S}$ и вектором действия $a \in \mathcal{A}$ (также иногда называемым управляющим сигналом). В качестве пространства состояний системы $\mathcal{S}$ и пространства управляющих сигналов $\mathcal{A}$ рассматриваются d_s и d_a – мерные области, порожденные произведением замкнутых отрезков

$$s \in \mathcal{S} = [s_{\min}^{(1)}, s_{\max}^{(1)}] \times \ldots \times [s_{\min}^{(d_s)}, s_{\max}^{(d_s)}],$$
$$a \in \mathcal{A} = [a_{\min}^{(1)}, a_{\max}^{(1)}] \times \ldots \times [a_{\min}^{(d_a)}, a_{\max}^{(d_a)}],$$

dw – стандартный броуновский шум (винеровский процесс). Даны функции *дрейфа* $b(s,a) : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^{d_s}$ и *диффузии* $\sigma_{ij}(s) : \mathcal{S} \to \mathbb{R}^{d_s^2}$. Для постановки задачи оптимального управления также зададимся *моментальной функцией награды* $r(s,a)$, зависящей от текущего состояния системы s и текущего управляющего сигнала a, а также коэффициентом дисконтирования β. Функции $b(s,a)$, $\sigma_{ij}(s)$ полагаются ограниченными на своих областях определений.

На функцию $r(s,a)$ накладывается следующее ограничение:

$$|r(s,a)| \le C(1 + |s|^k + |a|^k), \tag{2}$$

где $C \in \mathbb{R}_+$ и $k \in \mathbb{N}$ – некоторые константы.

Потребуем выполнения еще одного условия: матрично-значная функция диффузии $\sigma_{ij}(s)$ является диагональной ($\sigma_{ij} = 0$ если $i \neq j$). Такая ситуация имеет место, например, в случае присутствия нескоррелированного шума в датчиках, измеряющих текущее состояние системы s.

Ставится следующая задача: найти оптимальную детерминистическую политику (в зависимости от области науки также называемую оптимальным управлением, регулятором или стратегией) $\pi^*(s) : \mathcal{S} \to \mathcal{A}$. Оптимальность здесь подразумевается в следующем смысле: если данная политика используется для генерации сигнала управления в каждый момент времени $a(t) = \pi^*\big(s(t)\big)$, то матожидание суммарной награды (в общем случае с дисконтированием) по реализациям всех возможных случайных траекторий за все доступное время будет максимальным:

$$\pi^*(s) = \arg\max_a \left[\mathbb{E} \int_{t=0}^{T} \exp(-\beta t) r(s(t), a) dt \right], \tag{3}$$

где β – коэффициент дисконтирования в непрерывном времени, T – время конца эпизода (для дисконтированных нетерминальных задач управления $T = \infty$). Необходимым образом требуется, чтобы матожидание награды было ограничено: $\mathbb{E}\int_{t=0}^{T} e^{-\beta t} r(s(t), a) dt < \infty$.

3. АППРОКСИМАЦИЯ МАРКОВСКИМ ПРОЦЕССОМ ПРИНЯТИЯ РЕШЕНИЙ

Говорят, что задан *марковский процесс принятия решений* (МППР) в дискретном времени, если задан следующий кортеж:

$$(\mathcal{S}, \mathcal{A}, T, R, \gamma), \tag{4}$$

где заданы конечное множество допустимых значений состояний $\mathcal{S}$, конечное множество допустимых значений действий (управляющих сигналов, управляющих решений) $\mathcal{A}$, известны все (условные) вероятности перехода из любого состояния s в любое состояние s' через любое действие a: $T(s, a, s') = P(s' | s, a)$, а также функция награды для любого такого перехода $R(s, a, s')$ и коэффициент дисконтирования $\gamma \in (0, 1]$. Если $\gamma = 1$, то МППР называется *недисконтированным*.

Для численного нахождения решения исходной задачи стохастического оптимального управления (1) дискретизуем ее. Общая теория дискретизации для таких задач может быть найдена в книгах В. Флеминга (см. [2]) и Г. Кушнера (см. [1]). Основная идея этой теории в том, чтобы спроектировать непрерывные пространства состояний и действий на сетку, а затем построить МППР, который был бы эквивалентен нашему стохастическому процессу управления (1) в смысле дрифта и диффузии за единицу времени:

$$\lim_{h \to 0} \frac{\mathbb{E}[s_{t+1} - s_t \,|\, s_t = s,\ a_t = a]}{\Delta t} = b(s, a),$$

$$\lim_{h \to 0} \frac{\mathrm{Cov}[s_{t+1} - s_t \,|\, s_t = s,\ a_t = a]}{\Delta t} = \sigma(s, a).$$

Здесь $h = \min(h_i)$, а h_i – шаги дискретизации равномерной сетки вдоль i-й оси: $h_i = (s^{(i)}_{\max} - s^{(i)}_{\min})/(N^{(i)}_s - 1)$.

Мы будем использовать модифицированную версию схемы расщепления против потока, предложенной в [9], построенной по общей методике Г. Кушнера. Данная схема разрешает переходы с ненулевой вероятностью только между соседними узлами сетки, и при этом все диагональные переходы запрещены. С учетом того, что диффузионный член $\sigma_{ij}(s)$ диагонален, схема дискретизации принимает следующий вид:

$$Q^h = \max_{s,a} \left(\sum_i \frac{|b_i(s, a)|}{h_i} + \frac{\sigma_i^2(s)}{h_i^2} \right),$$

$$\Delta t^h = 1/Q^h,$$

$$b^+ = \begin{cases} b(s, a), & \text{если} \quad b(s, a) > 0, \\ 0 & \text{иначе}, \end{cases}$$

$$b^{-} = \begin{cases} -b(s,a), & \text{если} \quad b(s,a) < 0, \\ 0 & \text{иначе}, \end{cases} \tag{5}$$

$$T(s,a,s \pm e_i h_i) = \Delta t^h \left(\frac{b_i^{\pm}(s,a)}{h_i} + \frac{\sigma_i^2(s)}{2h_i^2} \right),$$

$$T(s,a,s) = 1 - \sum_{s'} T(s,a,s' \neq s),$$

$$R(s,a,s') = r(s,a)\Delta t^h,$$

$$\gamma = e^{-\beta \Delta t^h},$$

где $s \pm e_i h_i$ обозначает дискретное состояние (узел сетки), которое является соседним по отношению к узлу s вдоль i-й оси, в направлении увеличения или уменьшения индекса соответственно.

4. МАЛОРАНГОВОЕ РАЗЛОЖЕНИЕ В ТЕНЗОРНЫЙ ПОЕЗД

Рассмотрим d-мерный массив (тензор) $V(i_1, \ldots, i_d)$. Допустим, мы хотим найти его представление в следующем виде:

$$V(i_1, i_2, \ldots, i_d) = \sum_{m_1, \ldots, m_{d-1}}^{r_1, r_2, \ldots, r_{d-1}} G^{(1)}(i_1, m_1) G^{(2)}(m_2, i_2, m_2) \cdot \ldots \cdot G^{(d)}(m_{d-1}, i_d). \tag{6}$$

Такое разложение называется *разложением в тензорный поезд* или *TT-разложением*. Числа $r_1, \ldots, r_{d-1}$ называются *TT-рангами*. Чтобы равенство (6) выполнялось точно, необходимо хранить в худшем случае $O(dN^3)$ чисел из вместо $O(N^d)$. Зачастую имеет смысл найти такой тензорный поезд, что данное равенство соблюдается с некоторой погрешностью (в смысле среднеквадратичной ошибки), но TT-ранги малы ($\forall i : r_i < 100$). В таком случае можно ожидать, что функция на сетке будет сжата с требованием по памяти, равным $O(dNR^2)$, где $R = \max(r_i)$.

Важным фактом является то, что для достаточно широкого класса функций, спроецированных на равномерную сетку (более полное описание дано в [10]), если мы зафиксируем наибольшую допустимую ошибку аппроксимации, при измельчении сетки TT-ранги не будут расти, либо будут расти лишь логарифмически. Отметим, что в дополнение к сжатию в тензорных поездах сохраняется возможность выполнять алгебраические ($+$, $-$, $*$, $\otimes$, $\circ$) и линейно-алгебраические операции (внешние и внутренние произведения, свертки, суммирования по индексам), операции взятия подмассива и доступа к индивидуальным элементам, а также применять произвольные функции к тензорным поездам с помощью алгоритмов из семейства многомерной крестовой аппроксимации (см. [11]).

5. УРАВНЕНИЕ БЕЛЛМАНА

Метод нахождения глобально-оптимальной политики решений для марковского процесса принятия решений был предложен Р. Беллманом в свом классическом труде [12]. Рассмотрим его основные положения.

Зададимся МППР ($\mathcal{S}$, $\mathcal{A}$, T, R, γ). Задачей является найти такую управляющую политику, которая, стартуя из любого состояния s, максимизирует матожидание (возможно, дисконтированной) награды за все последующее время:

$$V(s) = \mathbb{E} \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1}) \to \max.$$

Величина $V(s)$ называется *функцией ценности* (value function) и является ключевым элементом исследований в стохастическом оптимальном управлении и науке о машинном обучении с подкреплением.

Один из главных результатов Беллмана заключался в следующем: глобально оптимальная политика (управление, регулятор) для МППР не зависит от предыстории, а зависит только от текущего состояния. Из этого следует, что для глобальной оптимальности политики достаточно, что-

бы действие было локально оптимально, но не в смысле текущего приращения функции награды $R(s,a)$, а в смысле текущего приращения функции ценности $V(s)$.

Рекуррентные соотношения значений функции ценности в соседних состояниях, разделенных всего одним временным шагом, называются *уравнениями Беллмана* и имеют вид

$$V^*(s) = \max_a \left[R(s,a) + \gamma \sum_{s'} V^*(s') T(s,a,s') \right]. \tag{7}$$

Для краткости мы использовали следующее обозначение:

$$R(s,a) = \mathbb{E}_{s'} R(s,a,s') = \sum_{s'} R(s,a,s') T(s,a,s').$$

Если рассмотреть правую часть уравнения (7) как оператор $\hat{\beta}$, действующий на функцию $V(s)$, то такой оператор в литературе называется *оператором оптимальности Беллмана* (Bellman Optimality Operator).

Уравнение на оптимальную политику выглядит следующим образом:

$$\pi^*(s) = \arg\max_a \left[R(s,a) + \gamma \sum_{s'} V^*(s') T(s,a,s') \right]. \tag{8}$$

Заметим, что ряд задач оптимального управления (называемых *терминальными*), например, задача оптимального быстродействия, требуют задания терминальной области. При попадании в терминальную область счетчик времени останавливается, и дальшейшее накопление награды невозможно. Чтобы математически ввести терминальную область $\mathcal{S}_{\text{terminal}} \subset \mathcal{S}$ на языке Беллмана, необходимо положить нулем все вероятности переходов для состояний из данной области, кроме переходов каждого состояния само в себя: $\forall (s,a) \in \mathcal{S}_{\text{terminal}} \times \mathcal{A} : T(s,a,s') = \delta_{ss'}$, где $\delta_{ss'}$ – символ Кронекера. Также функция награды должна быть положена нулем для всех переходов, начинающихся из терминальной области: $\forall s \in \mathcal{S}_{\text{terminal}} : R(s,a) = 0, V(s) = 0$.

5.1. Итерация функции ценности в формате тензорного поезда

Уравнение Беллмана является уравнением на неподвижную точку для нелинейного оператора оптимальности Беллмана $\hat{\beta}$, и оно может быть решено методом простых итераций. Такой подход был предложен самим Р. Беллманом в его классической работе [12] и называется методом *итерации функции ценности* (Value Iteration, VI). Он использует то обстоятельство, что оператор $\hat{\beta}$ является сжимающим отображением почти во всех практически применимых случаях (в случае дисконтированных задач, либо в случаях недисконтированных задач с терминальной областью, достижимой хотя бы одной политикой).

Для обоснования использования малорангового разложения рассмотрим количество занимаемой памяти для МППР, аппроксимирующего задачу стохастического оптимального управления (1). Поскольку используется равномерная сетка дискретизации вдоль каждой оси пространства состояний в N_s узлов, результирующая сложность по памяти для хранения оптимальной функции ценности $V^*(s)$ и оптимальной политики $\pi^*(s)$ будет составлять $O(N_s^{d_s})$ для каждой из этих функций. Даже простые мобильные робототехнические аппараты, такие как БПЛА самолетного типа или квадрокоптеры, задаются $d_s = 12$-мерным вектором состояния и минимум $d_a = 3$-мерным вектором управляющих сигналов. В итоге при достаточно грубой дискретизации сетки в 100 узлов по каждой координате в задаче возникает минимум $N_s^{d_s} = 10^{24}$ элементов, что делает прямое численное нахождение решения в беллмановском формализме невозможным. Если при поиске решения вероятности переходов $T(s,a,s')$ не вычисляются каждый раз, а хранятся, то дополнительно потребуется сохранить еще $(2d_s + 1)N_s^{d_s} N_a^{d_a}$ чисел (где N_a – число узлов дискретизации вдоль осей пространства управляющих сигналов, а d_a – размерность этого пространства).

В статье [7], предложенной в Массачуссетском технологическом институте А. Городетским, предлагалось хранить функцию ценности в виде малорангового тензорного поезда и применять оператор оптимальности Беллмана в рамках итерации функции ценности методом крестовой аппроксимации для тензорных поездов (см. [11]). В случае малых TT-рангов функции ценности

ее сложность по памяти понижается с $O(N^d)$ до $O(dNr^2)$, что делает задачу вычислительно решаемой даже на маломощном компьютере. Также в [7], [9] показано, что в TT-формате оператор $\hat{\beta}$ сохраняет свойство быть сжимающим отображением, и гарантии сходимости, выведенные для изначальной итерации функции ценности (см. [12]), сохраняются:

Data: $R(s,a), \gamma, T(s,a,s')$

Result: $V^*(s)$

while $\epsilon = \dfrac{\left\|V^{(k+1)} - V^{(k)}\right\|_2}{\left\|V^{(k)}\right\|_2} < tol$ **do**

$V^{k+1} = TTCROSS((7), \epsilon)$

$k = k + 1$

end

Алгоритм 1: TT-Value Iteration (TT-VI)

Комбинация тензорных разложений и беллмановской постановки задачи управления показала отличные результаты для квадратичных и терминальных задач управления для ряда нелинейных неаффинных систем управления, соответствующих различным малоприводным робототехническим платформам: обратному маятнику, машине Дубинса, планеру, квадрокоптеру (см. [6], [7]), в том числе при наличии стохастического воздействия в системе.

Для дальнейших рассуждений о сложности алгоритмов введем следующие обозначения:

$$R_V = \max \operatorname{rank} V(s,a),$$
$$R_P = \max_{i=0,1,\ldots d_s,\pm} \operatorname{rank} P_i^{\pm}(s,a).$$

В наивной имплементации TT-VI алгоритма крестовая аппроксимация требует $O(dNR_V^3)$ на каждое применение оператора оптимальности Беллмана, а также требуется $O(dR_V^2)$ операций для распаковывания значений из TT в каждой точке, что в итоге приводит к вычислительной сложности $O(d^2NR_V^5)$.

Однако вычисления в алгоритме TT-VI возможно оптимизировать. Для этого вспомним, что метод крестовой аппроксимации считывает значения функции в точках, которые сгруппированы в виде строк, столбцов и их многомерных обобщений (распорок) многомерной сетки. Оценим сложность такого алгоритма. Алгоритм крестовой аппроксимации требует $O(dR_V^3)$ распорок, а извлечение одной распорки из TT-разложения требует $O(dR_V^2)$ операций (что меньше, чем извлечение одного числа). Результирующая сложность алгоритма составляет $O(d^2R_V^5)$ операций. Далее в статье мы будем использовать именно эту оптимизированную версию как базовый алгоритм для сравнения.

5.2. Q-итерация в формате тензорного поезда

В данной статье мы предлагаем модификацию алгоритма TT-VI. Ее суть заключается в следующем: использовать стандарную итерацию функции ценности (алгоритм 1), но применять оператор оптимальности Беллмана в два этапа:

$$Q^k(s,a) = R(s,a) + \gamma \sum_{s'} T(s,a,s')V^k(s'), \tag{9}$$

$$V^{k+1} = \max_a Q^k(s,a). \tag{10}$$

Первый этап (9) включает в себя только стандартные тензорные операции, такие как сложения и суммирования по индексам, тогда как второй (10) требует взятия нелинейной функции (максимума), для чего мы используем крестовый метод, также как в TT-VI алгоритме. Так как у двух алгоритмов совпадают вторые этапы, мы опустим оценку сложности для них в сравнительном анализе.

Рассмотрим первый этап применения оператора $\hat{\beta}$, в частности суммирование по индексам с тензором T. Как следует из используемой нами дискретизации (5), из каждого состояния s возможны только переходы в соседние состояния вдоль каждой оси $s' = s \pm e_i h_i$. Это означает, что сумма вероятностей по s' состоит из $2d_s + 1$ слагаемых: два на каждую координатную ось (вероятность перехода в направлении увеличения или уменьшения на 1) и еще одно на вероятность остаться на месте s:

$$\sum_{s'} T(s,a,s')V(s') = P_0(s,a)V(s) + \sum_{\text{sign}\in\{+,-\}} \sum_{i=1}^{d_s} P_i^{\text{sign}}(s,a)V_i^{-\text{sign}}(s),$$

где P_i^{sign} и P_0 — вероятности, посчитанные с помощью схемы дискретизации (5), а $V_i^{-\text{sign}}(s) = \text{circshift}(V(s_1, \ldots, s_{d_s}), -\text{sign}, i)$ — функция ценности (на сетке), циклически сдвинутая вдоль i-й оси на 1 узел, в направлении $-\text{sign}$.

Предложение 1. *Циклический сдвиг по k-й координате функции на сетке, представленной в виде тензорного поезда, достигается циклическим сдвигом всего одного (k-го) тензорного ядра $G^{(k)}(m_k, i_k, m_{k+1})$. Он стоит $O(NR^2)$ операций и не меняет тензорный ранг.*

Доказательство. Из формулы TT-разложения видно, что при циклическом сдвиге любого из свободных индексов $i_k \to i_k \pm 1 \bmod N_k$ равенство сохраняется:

$$V(i_1, \ldots, i_k \pm 1, \ldots, i_d) = \sum_{m_1,\ldots,m_{d-1}}^{r_1,\ldots,r_{d-1}} G^{(1)}(i_1, m_1) \cdot \ldots \cdot G^{(k)}(m_k, i_k \pm 1 \bmod N_k, m_k) \cdot \ldots \cdot G^{(d)}(m_{d-1}, i_d).$$

Замена $G^{(k)}(m_2, i_k, m_2) \to G^{(k)}(m_2, i_k \pm 1 \bmod N_k, m_2)$ требует перемещения всех элементов массива $G^{(k)}$, которых насчитывается $r_k N_k r_{k+1}$ для всех ядер кроме первого и последнего. Для первого и последнего ядра потребуется $N_1 r_1$ и $N_d r_{d-1}$ элементов соответственно, что в общем случае оценивается как $O(NR^2)$. Очевидно, что такая операция не меняет размер ядра $G^{(k)}$, поэтому ранг r_k сохраняется.

Чтобы вычислить поэлементные произведения $P(s,a)V(s)$ в формате тензорного поезда, необходимо добавить в функцию ценности “лишние” измерения, добавив новые (единичные) ядра: $P(s_1 \ldots s_{d_s}, a_1 \ldots a_{d_a}) \circ ((V(s)) \otimes 1_{a_1} \ldots \otimes 1_{a_{d_a}})$.

Предложение 2. *Сложность первой половины* (9) *применения оператора $\hat{\beta}$ в TT-QI составляет*

$$O(d^2 N R_V^3 R_P^3).$$

Доказательство. Одно применение оператора оптимальности Беллмана в алгоритме TT-QI (до этапа применения алгоритма крестовой аппроксимации) использует следующие тензорные операции:

- $2d_s$ циклических сдвигов $V(s)$, каждый по $O(NR_V^2)$, что в итоге составляет $O(dNR_V^2)$,
- 1 акт добавления в $V(s)$ единичных ядер с лишними измерениями, что составляет $O(d_a)$,
- взятие $2d_s + 1$ поэлементных произведений от тензорных поездов, сложностью $O(dNR_P^2 R_V^2)$ каждый, что дает $O(d^2 NR_P^2 R_V^2)$ в сумме,
- взятие $2d_s + 1$ TT-SVD округлений, что дает $O(dNR_P^3 R_V^3)$ каждый, и $O(d^2 NR_P^3 R_V^3)$ в сумме.

Полная сложность первого этапа составляет в итоге

$$O(dNR_V^2 + d_a + d^2 NR_P^2 R_V^2 + d^2 NR_P^3 R_V^3) = O(d^2 NR_P^3 R_V^3). \tag{11}$$

В случае существенно малых рангов R_P тензоров вероятностей эта асимптотика более выгодна относительно $O(d^2 R_V^5)$ в алгоритме TT-VI. Тензоры вероятностей $P_{0\ldots d_s}(s,a)$ вычисляются только один раз и их ранги R_P постоянны во время итерационного процесса, что в итоге дает асимптотическую сложность $O(R_V^3)$ в случае TT-QI вместо $O(R_V^5)$ в случае TT-VI. В наших численных экспериментах это привело к существенному (в 5–6 раз) выигрышу в производительности.

Таблица 1. Сравнение производительности

Параметр	TT-VI (оптимизизированный) : время, с	TT-QI : время, с
3-маятник, квадратичная награда, целевая ошибка $\delta = 10^{-4}$	1959.9	536.7
3-маятник, задача оптимального быстродействия, целевая ошибка $\delta = 5 \times 10^{-4}$	6024.2	2116.8
3-машина, квадратичная награда, целевая ошибка $\delta = 2 \times 10^{-3}$	2952.2	293.4
4-машина, квадратичная награда, целевая ошибка $\delta = 5 \times 10^{-3}$	3061.8	1109.3

6. ЧИСЛЕННЫЕ ЭКСПЕРИМЕНТЫ

Чтобы показать верность оптимальных политик (регуляторов), полученных на базе уравнения Беллмана, следует рассмотреть достаточно сложную систему, для которой решение не может быть получено более простыми методами из теории управления (такие как PID или линейно-квадратичные регуляторы).

Данный раздел содержит графики сходимости к решению уравнения Беллмана, а также результаты прямых симуляций траекторий, полученных под управлением оптимальной политики. Для интегрирования стохастических дифференциальных уравнений по Ито и расчета траекторий мы используем схему высокого порядка точности (1.0), предложенную Росслером (см. [13]).

6.1. Система 1: непериодический обратный маятник

Рассмотрим модифицированную задачу обратного маятника, где требуется поставить маятник в верхнее положение с нулевой угловой скоростью:

$$s = [\phi, \dot{\phi}]^{\mathrm{T}}, \tag{12}$$

$$b(s,a) = \left[\dot{\phi}, a - \sin(\phi)\right]^{\mathrm{T}}, \tag{13}$$

$$\sigma(s) = \mathrm{diag}\left([10^{-2}, 10^{-2}]\right). \tag{14}$$

Первая модификация – существенно малая мощность управляющего момента на валу маятника (максимальный момент сил на вал, вызванный управляющим сигналом, равен лишь 30% от максимального момента сил тяжести). Такая постановка задачи делает невозможным для регулятора достижение верхней точки любым способом, кроме как с помощью эксплуатации резонанса системы. Вторая модификация – это использование непериодического маятника, что также делает задачу управления сложнее.

Границы областей состояний и действий определены как

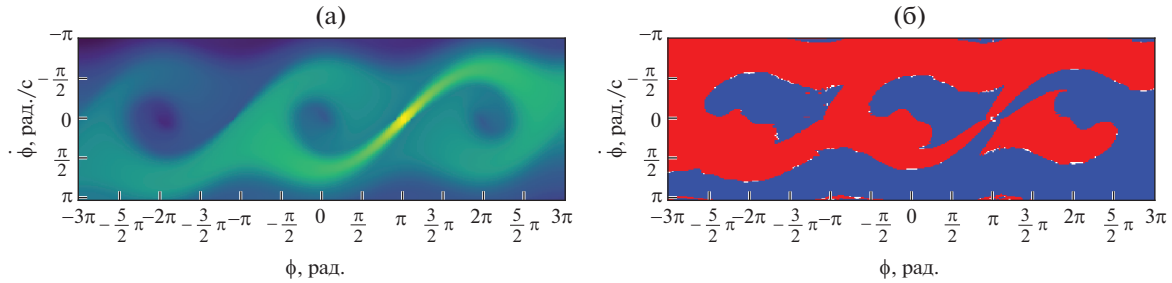
$$s \in [-3\pi, 3\pi] \times [-\pi, \pi], \tag{15}$$

$$a \in [-0.3, 0.3]. \tag{16}$$

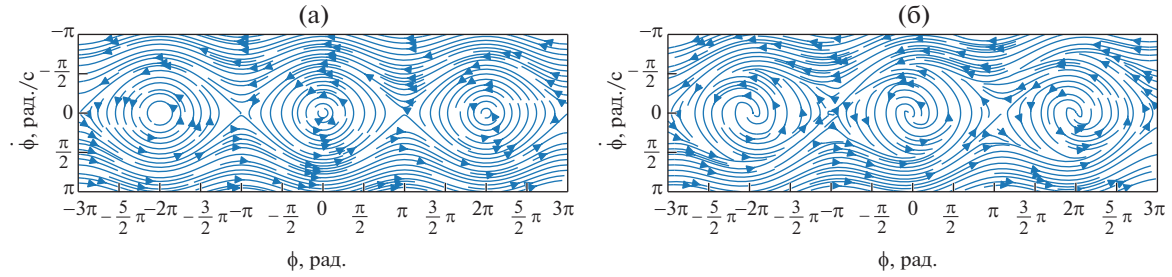
Мы рассмотрим решения двух вариантов данной задачи: с квадратичной наградой и задачу оптимального быстродействия.

6.1.1. Задача управления с квадратичной наградой. Задачи управления с квадратичной наградой/ценой хорошо изучены в литературе, поэтому в данной статье мы будем использовать такую постановку как тестовую для сравнения производительности (см. табл. 1).

Квадратичная награда задается следующим образом: $R_{QR}(s,a) = -(\phi - \pi)^2 - 0.8(\phi)^2 - 0.01a^2$ и коэффициент дисконтирования $\gamma = 0.999$.



Фиг. 1. Решение для задачи оптимального быстродействия непериодическим маятником: (а) – оптимальная функция ценности $V(s)$, (б) – оптимальный регулятор $\pi^*(s)$.



Фиг. 2. Фазовые портреты динамической системы непериодического маятника: (а) – без управления ($a \equiv 0$), (б) – под управлением оптимального регулятора ($a = \pi^*(s)$).

6.1.2. Задача оптимального быстродействия. Для того чтобы сформулировать задачу оптимального быстродействия на беллмановском языке, необходимо положить награду для каждого перехода ($s \xrightarrow{a} as'$) равной отрезку времени, которое требуется, чтобы этот переход совершить:

$$R(s,a,s') = \begin{cases} -\Delta t(s,a,s'), & \text{если} \quad s \notin \mathcal{S}_{\text{terminal}}, \\ 0, & \text{если} \quad s \in \mathcal{S}_{\text{terminal}}. \end{cases} \tag{17}$$

В таком случае функция ценности становится равна матожиданию времени прибывания в терминальную область:

$$V(s) = \sum_{t=0}^{\infty} \mathbb{E} R(s,a,s') = \sum_{t=0}^{\infty} \mathbb{E}\Delta t(s,a,s') = -\sum_{t=0}^{\tau} \mathbb{E}\Delta t = -\tau. \tag{18}$$

Терминальную область в данном случае мы полагаем малой окрестностью целевого состояния $(\pi, 0)$, что соответствует перевернутому маятнику с нулевой угловой скоростью:

$$\mathcal{S}_{\text{terminal}} = [\pi - 2h_0, \pi + 2h_0] \times [-2h_1, 2h_1]. \tag{19}$$

Коэффициент дисконтирования положен равным единице ($\gamma = 1$), т.е. решается недисконтированная задача.

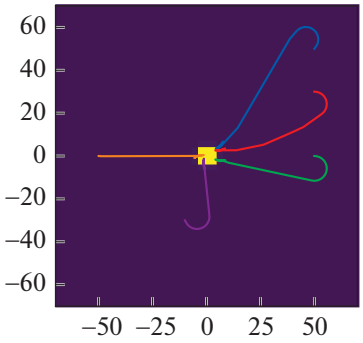
Решение данной задачи на сетке с дискретизацией $301 \times 151 \times 51$ дает оптимальную функцию ценности, оптимальный регулятор (см. фиг. 1) и соответствующим образом изменяет фазовый портрет динамической системы под действием регулятора, как показано на фиг. 2.

6.2. Система 2: машина Дубинса

Другим известным примером малоприводной системы является машина (автомобиль) Дубинса.

6.2.1. Простая машина Дубинса. Рассмотрим простейшую модель колесного автомобиля, описываемого уравнениями

$$s = [x, y, \phi]^{\mathrm{T}}, \tag{20}$$



Фиг. 3. Симулированные траектории (x, y) машины Дубинса с инерцией.

$$a = [u, \theta]^{\mathrm{T}}, \tag{21}$$

$$b(s,a) = \left[u\cos\phi,\ u\sin\phi,\ \frac{u}{L}\operatorname{tg}\theta\right]^{\mathrm{T}}, \tag{22}$$

$$\sigma(s) = \operatorname{diag}(10^{-3}, 10^{-3}, 10^{-3}). \tag{23}$$

Для усложнения маневра положим, что данная машина может ехать только вперед. Пространства состояний и действий тогда имеют следующий вид:

$$\mathcal{S} = [-100, 100]\times[-100, 100]\times SO(2),$$
$$\mathcal{A} = [0, 1]\times\left[-\frac{\pi}{3}, -\frac{\pi}{3}\right],$$

где $SO(2)$ – одномерная окружность. Функция награды задана следующим образом:

$$r(s,a) = -10^{-4}x^2 - 10^{-4}y^2 - 10^{-5}\phi^2 - 10^{-3}u^2.$$

Результаты тестов сходимости показаны ниже на фиг. 6а.

6.2.2. Машина Дубинса с инерцией. Данная модель машины является усложненной версией модели из п. 6.2.1. В ней сигнал управления не влияет на скорость напрямую, а лишь управляет ускорением. В этой модели автомобиль так же может ехать только вперед, а ускоряться может и вперед, и назад:

$$s = [x, y, v, \phi]^{\mathrm{T}}, \tag{24}$$

$$a = [u, \theta]^{\mathrm{T}}, \tag{25}$$

$$b(s,a) = \left[v\cos\phi,\ v\sin\phi,\ u,\ \frac{u}{L}\operatorname{tg}\theta\right]^{\mathrm{T}}, \tag{26}$$

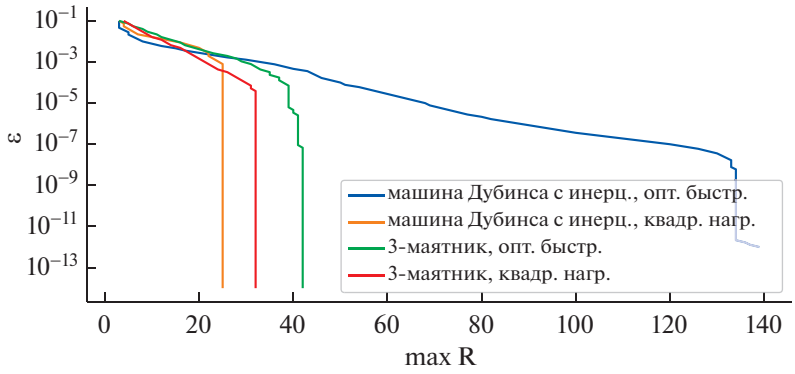
$$\sigma(s) = \operatorname{diag}(10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}), \tag{27}$$
$$\mathcal{S} = [-70, 70]\times[-70, 70]\times SO(2),$$
$$\mathcal{A} = [-2, 1]\times\left[-\frac{\pi}{3}, -\frac{\pi}{3}\right].$$

Мы рассмотрим два варианта награды для этой системы: (A) – для задачи оптимального быстродействия ($r = r_A$, $\gamma = 1$), (B) – для задачи с квадратичной наградой (r_B, $\gamma = 0.999$):

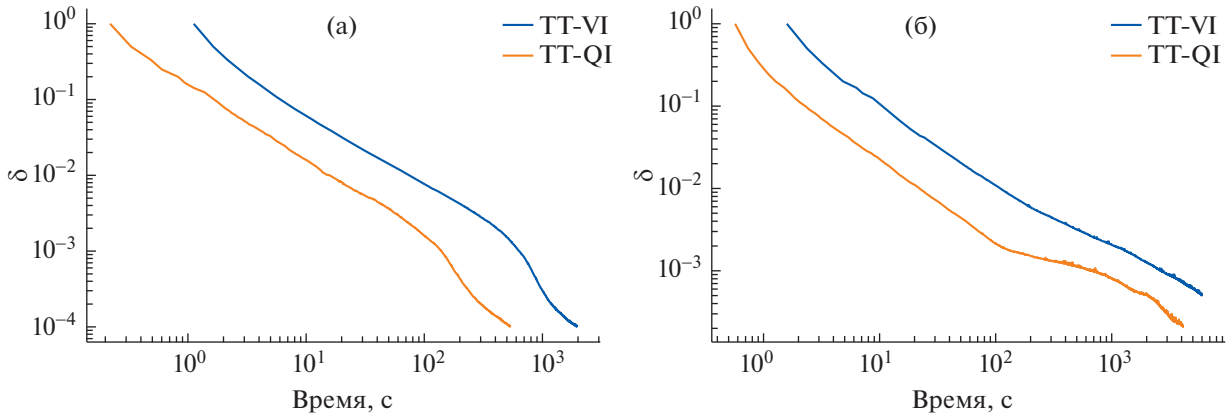
$$r_A(s,a) = -\Delta t(s,a), \tag{28}$$

$$r_B(s,a) = -10^{-4}x^2 - 10^{-4}y^2 - 10^{-5}\phi^2 - 10^{-3}u^2. \tag{29}$$

На фиг. 3 видно несколько симулированных траекторий координат (x, y) для задачи оптимального быстродействия машины Дубинса с инерцией (вариант A). Все траектории прибыли в



Фиг. 4. Сжимаемость функции ценности в разных задачах.



Фиг. 5. График относительной ошибки решения уравнения Беллмана для алгоритмов TT-VI и TT-QI: (а) – 3-маятник (квадратичная награда), (б) – 3-маятник (задача оптимального быстродействия).

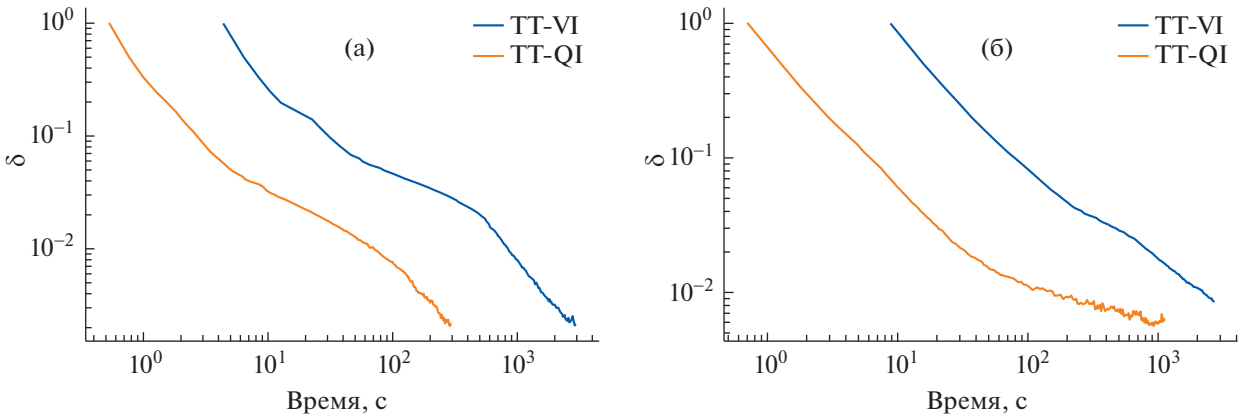
терминальную область. Также видно, что траектории близко соответствуют теоретически предсказаннной форме кривых (путям Дубинса).

7. СРАВНЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ

В данном разделе мы сравниваем производительность нашего алгоритма (TT-QI) с оптимизированной версией алгоритма TT-VI. На фиг. 4 изображена зависимость максимального ранга функций ценности R_V от целевой ошибки сжатия ε с помощью TT-SVD. Видно, что функции ценности для данных задач действительно с высокой точностью являются малоранговыми.

Для экспериментов, приведенных ниже, функция ценности инициализировалась нулевыми малоранговыми тензорами. Вычисления проводились на рабочей станции с 3.5 ГГц 8-ядерным процессором производства AMD и 12 Гб оперативной памяти. На фиг. 5 и 6 видно, что оба алгоритма ведут себя практически идентично и имеют степенной закон сходимости, но отличаются по затраченному времени на константный множитель.

Важно отметить, что графики сходимости обоих алгоритмов от числа итераций ведут себя идентично, так как аппроксимируют один и тот же оператор оптимальности Беллмана с достаточно высокой точностью. А скорость сходимости итерации функции ценности зависит только от размера пространства действий и скорости перемешивания в марковской цепи, которые совпадают.



Фиг. 6. График относительной ошибки решения уравнения Беллмана для алгоритмов TT-VI и TT-QI: (а) – простая машина Дубинса (квадратичная награда), (б) – машина Дубинса с инерцией (квадратичная награда).

На фиг. 5 и 6 построена относительная ошибка δ, которая для k-й итерации определяется как

$$\delta_k = \frac{\|V_k - V_{k-1}\|_2}{\|V_{k-1}\|_2}. \tag{30}$$

Сравнение времени решения с относительной ошибкой δ показано в табл. 1.

8. ЗАКЛЮЧЕНИЕ

Мы предложили модифицированный алгоритм для итерации функции ценности для задач стохастического оптимального управления в формате малорангового тензорного поезда. Если TT-ранги функции вероятностей перехода (после применения схемы расщепления) малы, а также TT-ранги функции награды малы, наш алгоритм имеет меньшую вычислительную сложность, чем существующий метод решения стохастического оптимального управления для общего случая неаффинных систем, существующий в литературе (см. [7]).

В численных экспериментах на примере классических нелинейных малоприводных задач управления (непериодический обратный маятник, машины Дубинса) для задачи с квадратичной наградой и задачи оптимального быстродействия наш метод позволил сократить время нахождения точного решения вплоть до 10 раз.

Так как данный подход позволяет решать задачи управления для систем достаточно большой размерности и общего вида, данный результат может быть важен для синтеза сложных движений и маневров в различных сферах робототехники.

Авторы благодарны С. Долгову (университет Бата), С. Матвееву (ИВМ РАН, Сколтех) и Г. Овчинникову (Сколтех) за ценные обсуждения.

СПИСОК ЛИТЕРАТУРЫ

1. *Kushner H., Dupuis P.G.* Numerical methods for stochastic control problems in continuous time. Springer, 2013, V. 24.
2. *Fleming W.H., Soner H.M.* Controlled Markov Processes and Viscosity Solutions. Springer, 2006.
3. *Bertsekas D.P., Tsitsiklis J.N.* Neuro-Dynamic Programming, 1st ed. Athena Scientific, 1996.
4. *Lillicrap T.P. et al.* Continuous control with deep reinforcement learning // 4th Inter. Conf. Learn. Represent. ICLR, 2016.
5. *Kidzinski et al.* Learning to run challenge solutions: Adapting reinforcement learning methods for neuromusculoskeletal environments // Proceed. NIPS, 2017.
6. *Horowitz M., Damle A., Burdick J.W.* Linear Hamilton-Jacobi-Bellman equations in high dimensions // IEEE Conf. Decis. Control, 2014.
7. *Gorodetsky A.A., Karaman S., Marzouk Y.M.* Efficient high-dimensional stochastic optimal motion control using Tensor Train decomposition // Robotics: Sci. Syst. 2015.

8. *Gorodetsky A.A., Karaman S., Marzouk Y.M.* High-dimensional stochastic optimal control using continuous tensor decompositions // Inter. J. Robot. Res. 2018. № 37. Iss. 2–3.
9. *Tal E., Gorodetsky A., Karaman A.* Continuous Tensor Train-based dynamic programming for high-dimensional zero-sum differential games // Am. Control Conf. 2018.
10. *Oseledets I.V., Tyrtyshnikov E.E.* Breaking the curse of dimensionality, or how to use SVD in many dimensions // SIAM J. Sci. Comp. 2009. V. 31. № 5. P. 3744–3759.
11. *Oseledets I.V., Tyrtyshnikov E.E.* TT-cross approximation for multidimensional arrays // Lin. Alg. Appl. 2010. V. 432. № 1. P. 70–88.
12. *Bellman R.E.* Dynamic programming. Princeton Univ. Press, 1957.
13. *Rossler A.* Runge–Kutta methods for the strong approximation of solutions of stochastic differential equations // SIAM J. Numer. Anal. 2010. V. 3. № 48. P. 922–952.