Математическое моделирование движения руки и поведенческих движений

студент 2 курса магистратуры К. Ю. Егоров
научный руководитель — к.ф-м.н., доцент И. В. Востриков

Кафедра системного анализа
ВМК МГУ

5 апреля 2023 г.

Введение

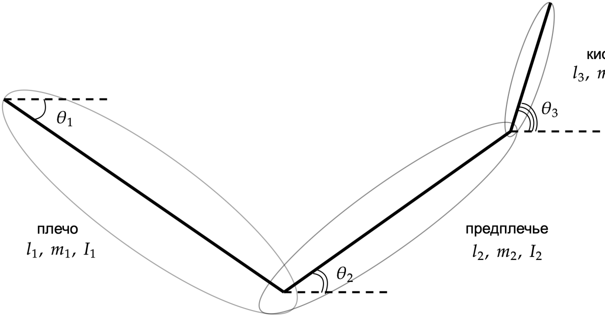
В работе приведен анализ движения руки человека:

- Построена математическая модель
- Поставлена задача оптимального управления для достижения целевого положения руки
- Предложен итеративный алгоритм синтеза управления

Возможные применения

- Разработка *умных* протезов
- Разработка устройств для управления, корректирования движений людей с болезнями нервной системы

Математическое моделирование



Начальные данные

- Рука — 3-х сочленённый математический маятник.
- Известны длины ℓ_i, массы m_i и моменты инерции I_i.
- Фазовые переменные — углы поворота сочленения θ_i относительно оси Ox.

Метод Эйлера–Лагранжа

$$\mathcal{L} = \Pi - K \implies \tau_i = \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}_i}\right) - \frac{\partial \mathcal{L}}{\partial \theta_i},$$

где K и Π — общие кинетическая и потенциальная энергии системы.

Уравнение динамики

$$\tau = M(\theta)\ddot{\theta} + L(\theta, \dot{\theta})$$

- τ_i — момент силы, действующей на i-е сочленение
- $M(\theta) = M^{\mathrm{T}}(\theta) > 0$ — матрица инерции
- $L(\theta, \dot{\theta})$ — вектор центростремительных и корелисовых сил

Математическое моделирование

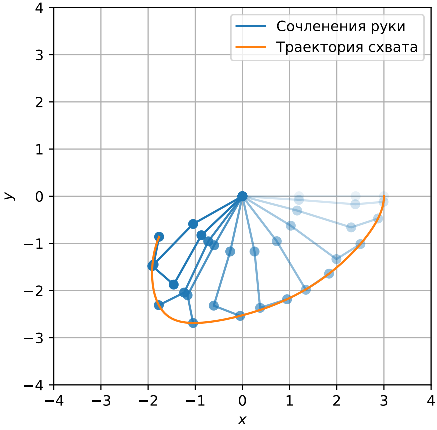


Рис.: Траектория руки в свободном падении

Принцип оптимальности

- Биологическое движение требует обработки большого количества информации
- Моторно-двигательная система как результат эволюции и обучения строит движение в соотвествии с принципом оптимальности

Биологический принцип оптимальности

Выбираемые нервной системой схемы движения являются оптимальными для поставленной задачи.

Энергетические затраты

В работе [2] показано, что оптимизации проводится с целью уменьшения затрат энергии.

Формализация энергетических затрат [3]

$$\text{Затраты} = \int_{t_{start}}^{t_{final}} \|\dot{\tau}\|^2 \, dt$$

Задача достижения целевого положения

Введем $x = [\theta\ \dot{\theta}\ \tau]^{\mathrm{T}}$, тогда уравнение динамики примет вид

$$\dot{x} = A(x) + Bu, \text{ где } A(x) = \begin{bmatrix} x_2 \\ M^{-1}(x_1)(x_3 - L(x_1, x_2)) \\ 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

Задано начальное положение:

$$x(t_{start}) = x^{start}$$

Задача минимизации функционала:

$$J = w_1 \underbrace{\langle x - x^{final}, x - x^{final} \rangle}_{Q^{final}(x)} + w_2 \int\limits_{t_{start}}^{t_{final}} \underbrace{\langle u, u \rangle}_{Q(x,u)} dt \longrightarrow \min$$

Дискретизация задачи

Введем сетку по переменной $t = \{t_k\}_{k=1}^{N+1}$ с шагом Δt и дискретизируем задачу:

$$\begin{cases} x^{k+1} = f(x^k, u^k) \\ x^0 = x^{start}, \end{cases}$$

где $f(x^k, u^k) = \Delta t[A(x^k) + Bu^k] + x^k$.

$$J = Q^{N+1}(x^{N+1}) + \sum_{k=1}^{N} Q^k(x^k, u^k) \longrightarrow \min,$$

где $Q^k(x^k, u^k) = w_2 \Delta t Q(x^k, u^k)$, $Q^{N+1}(x^{N+1}) = w_1 Q^{final}(x^{N+1})$.

Идея метода

- Берётся некоторое *референсное* допустимое управление u и соответствующая ей референсная траектория x
- Задача управления решается точно на каждой итерации алгоритма для коррекции референсного управления с целью снижения значения функции цены
- Используя информацию о градиенте гамильтониана H на референсной траектории строится поправка вида

$$\Delta u = \eta \nabla_u H(u)$$

- Если решение не удовлетворяет заданной точности ε, получившееся управление берется в качестве референсного для следующей итерации алгоритма

Дифференциальное динамическое программирование

Пусть (u, x) — управление и соответствующая ему траектория на предыдущем шаге алгоритма. Рассмотрим Гамильтониан:

$$H = Q_{N+1}(x^{N+1}) + \sum_{k=0}^{N} Q_k(x^k, u^k) + \sum_{k=0}^{N} \lambda_k^{\mathrm{T}}(x^{k+1} - f(x^k, u^k))$$

Необходимые условия оптимальности 1-ого порядка

$$\frac{\partial H}{\partial \lambda_k} = x_{k+1} - f(x_k, u_k) = 0$$

$$\frac{\partial H}{\partial x^k} = \frac{\partial Q_k}{\partial x}(x^k, u^k) - \frac{\partial f}{\partial x}^{\mathrm{T}}(x^k, u^k)\lambda_k + \lambda_{k-1} = 0$$

$$\frac{\partial H}{\partial x^{N+1}} = \frac{\partial Q_{N+1}}{\partial x}(x^{N+1}) + \lambda_N = 0$$

$$\frac{\partial H}{\partial u^k} = \frac{\partial Q_k}{\partial u}(x^k, u^k) - \frac{\partial f}{\partial u}^{\mathrm{T}}(x^k, u^k)\lambda_k = 0$$

1. Рассчитать λ_k для $k = N,\, N-1,\, \dots,\, 1$:
$$\lambda_N = -\frac{\partial Q_{N+1}}{\partial x}(x^{N+1})$$
$$\lambda_{k-1} = \frac{\partial f}{\partial x^k}^{\mathrm{T}}(x^k, u^k)\lambda_k - \frac{\partial Q_k}{\partial x}(x^k, u^k)$$
2. Рассчитать $\frac{\partial H}{\partial u_k}$ для $k = N,\, N-1,\, \dots,\, 1$:
$\frac{\partial H}{\partial u_k} = \frac{\partial Q_k}{\partial u}(x^k, u^k) - \frac{\partial f}{\partial u}^{\mathrm{T}}(x^k, u^k)\lambda_k$
3. Сделать поправку на исходную траекторию u_k, прибавив к ней поправку $\Delta u_k = -\eta \frac{\partial H}{\partial u_k}$, где коэффициент η выбирается таким образом, чтобы получившееся управление было допустимым, то есть $u_k + \Delta u_k \in \mathcal{U}$.
4. Остановить алгоритм в случае, если $\sum_{k=0}^{n} \left\| \frac{\partial H}{\partial u_k} \right\|^2 < \varepsilon$.

Дифференциальное динамическое программирование

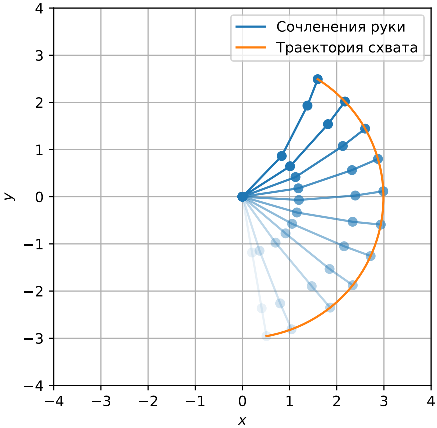


Рис.: Траектория руки, построенная методом ДДП.

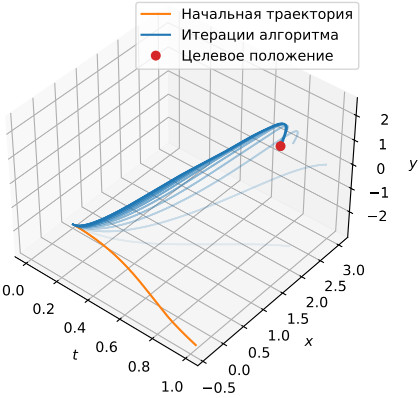


Рис.: Траектории схвата для итеративного алгоритма при нулевом начальном управлении $u^{ref} \equiv 0$. Показана каждая 3-я итерация.

Мотивация

- Градиентный метод сходится довольно долго
- Мы хотим выбрать начальное референсное управление таким образом, чтобы оно частично удовлетворяло условию задачи, с целью сократить число итераций алоритма
- Это управление должно *быстро* строиться
- Оно должно быть допустимым для рассматриваемой задачи

Референсная траектория

Приведем систему к линейной, заменой управления

$$v = M^{-1}(x_1)(\tau - L(x_1, x_2))$$

Тогда система примет вид:

$$x^{k+1} = \underbrace{\mathrm{diag}\{I, O\}}_{A_{ref}} x^k + \underbrace{\mathrm{diag}\{O, I\}}_{B_{ref}} v^k \tag{1}$$

Решим для системы (1) задачу минимизации интегрально-квадратичного функционала:

$$J^{ref} = \langle (x - x^{final}), T^{N+1}(x - x^{final}) \rangle + \sum_{k=1}^{N} \langle v^k, N v^k \rangle \longrightarrow \min$$

Получим референсное управление u из соотношения

$$\tau^k = M(x_1^k) v^k + L(x_1^k, x_2^k) \implies u^k = \frac{\tau^{k+1} - \tau^k}{\Delta t}$$

Референсная траектория

Метод динамического программирования даёт решение данной задачи:

$$v_*^k = -[N^k + B_{ref}^{\mathrm{T}} P^k B_{ref}]^{-1} P^k B_{ref} A_{ref} x,$$

где матрица P^k может быть посчитана в обратном времени как решение уравнения Риккати:

$$P^{k-1} = A_{ref}^{\mathrm{T}} P^k A_{ref} - A_{ref}^{\mathrm{T}} P^k B_{ref} [N + B_{ref}^{\mathrm{T}} P^k B_{ref}]^{-1} B_{ref}^{\mathrm{T}} P^k A_{ref}^k$$

$$P^{N+1} = T$$

Полученное управление

- Считается за одну итерацию
- Минимизирует изменение углового ускорения, значит, допустимо
- Приводит нас к целевому положению, но ничего не говорит об энергии

Референсная траектория

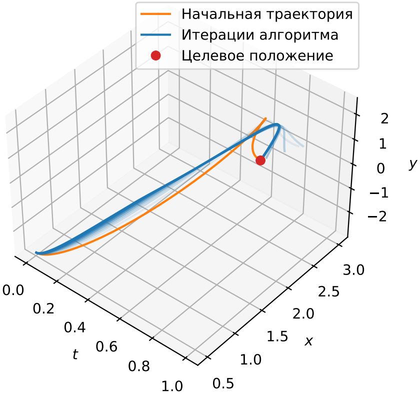


Рис.: Траектории схвата для итеративного алгоритма при новом начальном управлении. Показана *КАЖДАЯ* итерация.

Пример [Обход препятствий]

Добавим компонент в функцию цены:

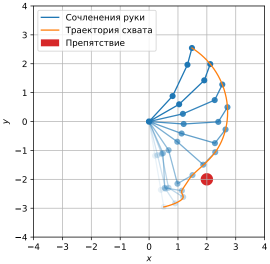
$$J_{\text{преп}} = w_3 \int_{t_{start}}^{t_{final}} \left(\left\| e_{\text{схв}} - e_{\text{преп}} \right\|^2 - r_{\text{преп}} \right)^{-2} dt,$$

где

- $e_{\text{схв}}$ — положение схвата
- $e_{\text{преп}}$ — центр препятствия
- $r_{\text{преп}}$ — радиус препятствия

Далее будем пользоваться алгоритмом.

Пример [Обход препятствий]



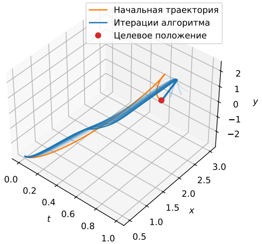


Рис.: Траектории руки и схвата для задачи обхода препятствия

Список литературы

1 Колюбин С. А. Динамика робототехнических систем // Учебное пособие. — СПб.: Университет ИТМО, 2017. — 117 с.

2 E. Todorov, M. Jordan. Optimal feedback control as a theory of motor coordination // Nature Neuroscience, Vol.5, No.11, 1226-1235, 2002.

3 Y. Uno, M. Kawato, R. Suzuki. Formation and control of optimal trajectory in human multijoint arm movement — minimum torque-change model // Biological Cybernetics 61, 89-101, 1989.

4 B.D.O. Anderson, J.B. Moore. Optimal Control: Linear Quadratic Methods // Prentice Hall, Upper Saddle River, 1990.

5 D. H. Jacobson. Differential dynamic programming methods for determining optimal control of non-linear systems // University of London, 1967.

6 E. Guechi, S. Bouzoualegh, Y. Zennir, S. Blažič. MPC Control and LQ Optimal Control of A Two-Link Robot Arm: A Comparative Study // Machines 6, no. 3: 37, 2018.

7 A. Babazadeh, N. Sadati. Optimal control of multiple-arm robotic systems using gradient method // IEEE Conference on Robotics, Automation and Mechatronics, Singapore, pp. 312-317 vol.1, 2004.