

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

Кафедра Систем управления и информатики

Дисциплина: Методы управления для робототехнических приложений (м.1.4.3-СУиИ)

## **ЛАБОРАТОРНАЯ РАБОТА 3:**

### **ПЛАНИРОВАНИЕ ТРАЕКТОРИИ ДВИЖЕНИЯ**

Студент: Дема Н.Ю.  
Группа: Р4135

## Задание

Для манипуляционного робота последовательной кинематики требуется по 4 точкам в операционном пространстве получить траекторию посредством интерполяции полиномами 4, 3 и 4 степени соответственно для каждого промежутка, разработать программу реализующую полученные уравнения и провести ее экспериментальную апробацию.

## Алгоритм построения траектории

Прежде чем приступить к планированию траектории, для каждой из 4-х заданных точек (a, b, c, d) определяется конфигурация манипулятора посредством решения обратной задачи кинематики, описываемой в предыдущей лабораторной работе. В результате получаются четыре вектора обобщенных координат  $\theta^{tr} = [\theta^a \ \theta^b \ \theta^c \ \theta^d]$

Затем для каждой обобщенной координаты определяются начальные условия, определяется количество искомых параметров (обобщенные координаты, скорости ускорения и т.д.) и выбираются соответствующие степени интерполирующих полиномов для построения траекторий.

Для задания многосегментной траектории воспользуемся нормированным временем для каждого из сегментов:

$$\tau_a = \frac{t - t_0}{t_1 - t_0}, \quad \tau_b = \frac{t - t_1}{t_2 - t_1}, \quad \tau_c = \frac{t - t_2}{t_3 - t_2} \quad (1)$$

где  $t_0, t_1, t_2, t_3$  — моменты времени прохождения полученных конфигураций.

Запишем начальные условия:

$$\dot{\theta}_a(1) = \dot{\theta}_b(0), \quad \ddot{\theta}_a(1) = \ddot{\theta}_b(0), \quad \dot{\theta}_b(1) = \dot{\theta}_c(0), \quad \ddot{\theta}_b(1) = \ddot{\theta}_c(0). \quad (2)$$

Выберем следующие полиномы для описания изменения обобщенных координат, скоростей и ускорений на каждом из сегментов:

$$\theta_a(\tau_a) = a_4\tau_a^4 + a_3\tau_a^3 + a_2\tau_a^2 + a_1\tau_a + a_0, \quad (3.1)$$

$$\theta_b(\tau_b) = b_3\tau_b^3 + b_2\tau_b^2 + b_1\tau_b + b_0, \quad (3.2)$$

$$\theta_c(\tau_c) = c_4\tau_c^4 + c_3\tau_c^3 + c_2\tau_c^2 + c_1\tau_c + c_0, \quad (3.3)$$

$$\dot{\theta}_a(\tau_a) = 4a_4\tau_a^3 + 3a_3\tau_a^2 + 2a_2\tau_a + a_1, \quad (4.1)$$

$$\dot{\theta}_b(\tau_b) = 3b_3\tau_b^2 + 2b_2\tau_b + b_1, \quad (4.2)$$

$$\dot{\theta}_c(\tau_c) = 4c_4\tau_c^3 + 3c_3\tau_c^2 + 2c_2\tau_c + c_1, \quad (4.3)$$

$$\ddot{\theta}_a(\tau_a) = 12a_4\tau_a^2 + 6a_3\tau_a + 2a_2, \quad (5.1)$$

$$\ddot{\theta}_b(\tau_b) = 6b_3\tau_b + 2b_2, \quad (5.2)$$

$$\ddot{\theta}_c(\tau_c) = 12c_4\tau_c^2 + 6c_3\tau_c + 2c_2, \quad (5.3)$$

Соотнося выражения (3-5) с начальными условиями, составим итоговую систему уравнений в матричном виде:

$$\theta_{tr0} = A [\mathbf{a}^T \ \mathbf{b}^T \ \mathbf{c}^T]^T \quad (6)$$

где соответствующие элементы раскрываются, как:

$$\begin{bmatrix} \theta_a(0) \\ \dot{\theta}_a(0) \\ \ddot{\theta}_a(0) \\ \theta_a(1) \\ \theta_a(1) \\ 0 \\ 0 \\ \theta_b(1) \\ \theta_b(1) \\ 0 \\ 0 \\ \theta_c(1) \\ \dot{\theta}_c(1) \\ \ddot{\theta}_c(1) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 4 & 3 & 2 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 12 & 6 & 2 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 & 2 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 6 & 2 & 0 & 0 & 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 3 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 12 & 6 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_4 \\ a_3 \\ a_2 \\ a_1 \\ a_0 \\ b_3 \\ b_2 \\ b_1 \\ b_0 \\ c_4 \\ c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix} \quad (7)$$

Определяя соответствующие коэффициенты **a**, **b** и **c** и подставляя время в полиномы (3-5) можно получить значения обобщенных координат, скоростей и ускорений для всей траектории.

## Результаты моделирования

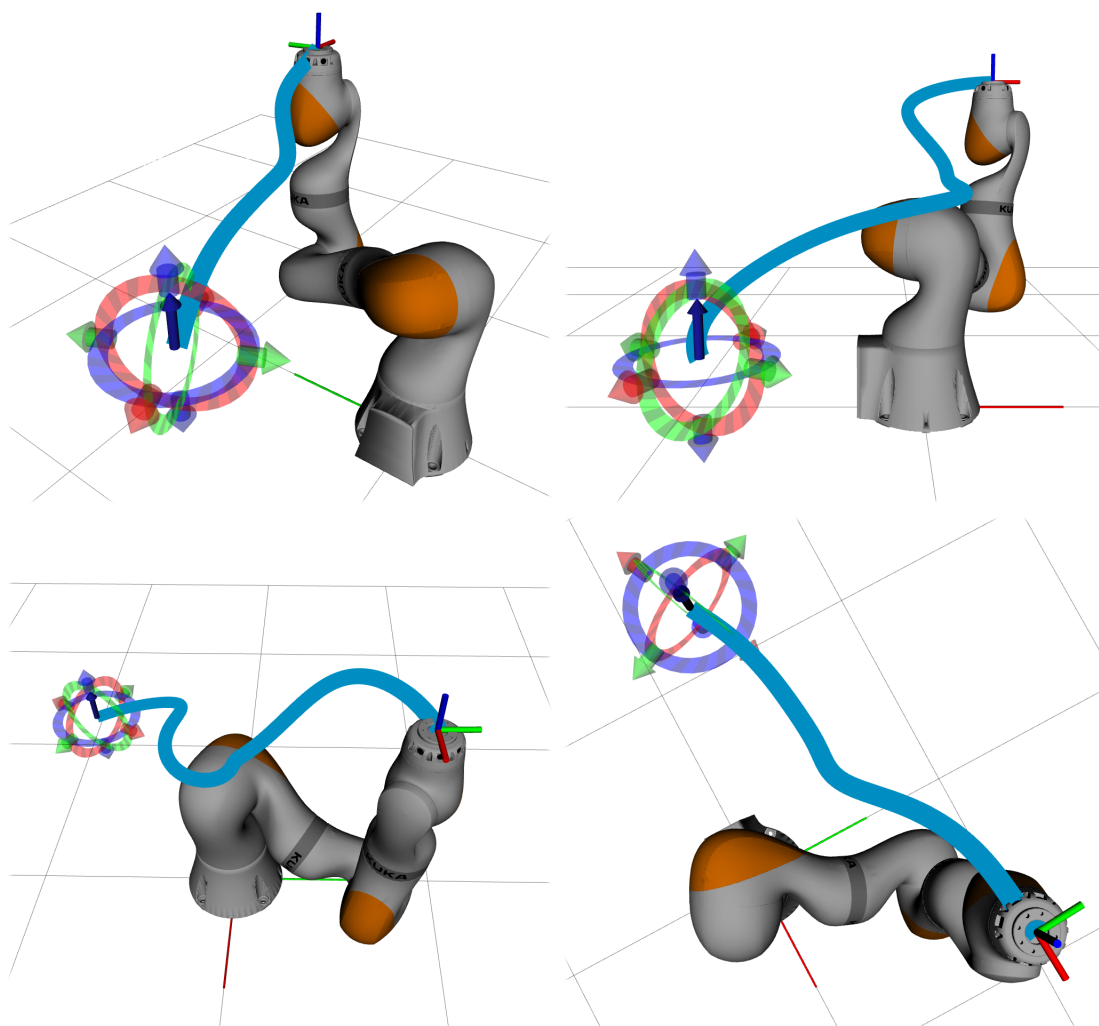


Рисунок 1: Траектории рабочего органа манипулятора

Для проведения расчетов воспользуемся средствами пакета Matlab/Simulink, соответствующий листинг программы представлен в Приложении 1.

Для визуализации результатов воспользуемся средствами r-viz. На Рисунке 1 представлены результаты расчета траектории рабочего органа для нескольких случаев.

## Заключение

В ходе выполнения данной работы был изучен метод построения траекторий посредством интерполирующих полиномов и успешно опробован на примере 6-го звенного манипулятора последовательной кинематики. Код разработанной программы доступен по адресу <https://github.com/Ram2301/iiwa>. Видео движения манипулятора по траекториям доступно по адресу: <https://youtu.be/2EVu1HQnMls>

## Приложение 1

---

```
1 % Trajectory generation by polinomial functions of 4-3-4 coresponding degrees
2
3 function tr = traj434(in_t, in_qs, in_q2, in_q3, in_qf, t)
4
5     n = 6;%numrows(in_qs);
6
7     %if nargin == 6
8         in_dqs = zeros(1,n)'; in_dqf = zeros(1,n)';
9         in_ddqs = zeros(1,n)'; in_ddqf = zeros(1,n)';
10    %end
11
12    %itr_n = in_t*100;
13    %t_step = 1/itr_n;
14
15    Am = [0  0  0  0  1  0  0  0  0  0  0  0  0  0;...
16          0  0  0  1  0  0  0  0  0  0  0  0  0  0;...
17          0  0  2  0  0  0  0  0  0  0  0  0  0  0;...
18          1  1  1  1  1  0  0  0  0  0  0  0  0  0;...
19          0  0  0  0  0  0  0  0  1  0  0  0  0  0;...
20          4  3  2  1  0  0  0  -1  0  0  0  0  0  0;...
21          12 6  2  0  0  0  -2  0  0  0  0  0  0  0;...
22          0  0  0  0  0  1  1  1  1  0  0  0  0  0;...
23          0  0  0  0  0  0  0  0  0  0  0  0  0  1;...
24          0  0  0  0  0  3  2  1  0  0  0  0  -1  0;...
25          0  0  0  0  0  6  2  0  0  0  0  -2  0  0;...
26          0  0  0  0  0  0  0  0  0  1  1  1  1  1;...
27          0  0  0  0  0  0  0  0  0  4  3  2  1  0;...
28          0  0  0  0  0  0  0  0  0  12 6  2  0  0];
29    Ami = inv(Am);
30
31    A = ones(6,14);
32    for i=1:n
33        A(i,:) = Ami*[in_qs(i); in_dqs(i); in_ddqs(i); in_q2(i); in_q2(i);0; 0; ...
34                      in_q3(i); in_q3(i); 0; 0; in_qf(i); in_dqf(i); in_ddqf(i)];
35    end
36
37    tr = zeros(1,n);
38    if t <= 1
39        for i=1:n
40            tr(1,i) = polyval(A(i,1:5),t);
41        end
42    end
43
44    if (t > 1) && (t <= 2)
45        for i=1:n
46            tr(1,i) = polyval(A(i,6:9),(t-1));
47        end
48    end
49
50    if (t > 2)
51        for i=1:n
52            tr(1,i) = polyval(A(i,10:14),(t-2));
53        end
54    end
55 end
```

---