

**Цель работы:** разработать программное обеспечение в Matlab для решения прямой (ПЗК) и обратной задачи кинематики (ОЗК) для робота-манипулятора ТУР-10К.

### Краткие теоретические сведения

Для решения ПЗК по заданному  $n$ -мерному вектору изменяемых параметров сочленений, таких, как углы или поступательные перемещения, в форме  $\theta_i = [\theta_1 \ \theta_2 \ \theta_3 \dots \theta_n]$ , необходимо получить соответствующую матрицу положения, связанную с положением и ориентацией схвата робота.

Для решения задач кинематики таких манипуляторов как ТУР-10К строится модель кинематики, предполагающая, что относительный угол поворота  $i$ -го звена  $q_i$  не зависит от углов поворота других звеньев. Механика робота ТУР-10К нарушает это предположение наличием влияния  $q_2$  на  $q_3$  и  $q_4$  на  $q_5$ .

Для отображения влияния  $q_2$  на  $q_3$  добавим в модель виртуальное сочленение  $q_3^v = -q_2$ , которое будет находиться в кинематической цепи между  $q_2$  и  $q_3$ , и будет совпадать с  $q_3$  по положению и при нулевой ориентации. Аналогично для отображения влияния  $q_4$  на  $q_5$  в модели кинематики добавим виртуальное сочленение  $q_5^v = -2q_4$  между  $q_4$  и  $q_5$ , которое будет совпадать с  $q_4$ . Таким образом, модель кинематики будет состоять из следующей последовательности сочленений:  $q_1 \ q_2 \ q_3^v \ q_3 \ q_4 \ q_5^v \ q_5$ .

Для решения ОЗК необходимо по заданной матрице положения определить вектор изменяемых параметров сочленений. Решений ОЗК может быть бесконечное множество.

В настоящей работе ПЗК решается методом Денавита-Хартенберга, ОЗК – генетическим алгоритмом.

### Ход работы

1. Известно, что положение и ориентация твердого тела (или системы координат, связанной с этим телом) в пространстве однозначно определяется шестью координатами: тремя линейными (декартовыми) и тремя угловыми (например, углами Эйлера). Использование метода, предложенного в 1955 г. учеными Жаком Денавитом и Ричардом Хартенбергом, позволяет сократить это число до четырех параметров, называемыми параметрами Денавита-Хартенберга. Такое упрощение достигается с помощью стандартизированного алгоритма привязки систем координат к звеньям манипулятора.

Параметры Денавита-Хартенберга:

$a_i$  – расстояние вдоль оси  $x_i$  от  $Z_{i-1}$  до  $Z_i$ ;

$\alpha_i$  – угол вокруг оси  $x_i$  от  $Z_{i-1}$  до  $Z_i$ ;

$d_i$  – расстояние вдоль оси  $Z_{i-1}$  от  $x_{i-1}$  до  $x_i$ ;

$\theta_i$  – угол вокруг оси  $Z_{i-1}$  от  $x_{i-1}$  до  $x_i$ ;

Решение прямой задачи кинематики методом Денавита-Хартенберга:

### Листинг программы

```
clear all;
teta(1)= 0; teta(2)= 45; teta(4)=0 ; teta(5)= 0; teta(7)= 0; % Угол м/у x_i и x_{i-1}
teta(3)=-teta(2);
teta(6)=-2*teta(5);
teta(2)=teta(2)-90;
teta(4)=teta(4)+90;
teta=teta*pi/180;

alp = [-90 0 0 0 90 0 0]*pi/180; % Угол м/у z_i и z_{i-1}
a = [0 500 0 670 220 0 0]; % Кратчайшее расстояние м/у z_i и z_{i-1}
d = [700 0 0 0 0 0 0]; % Кратчайшее расстояние м/у x_i и x_{i-1}

[A, T, coords] = den_hart(teta,alp, a, d);
disp('Координаты схвата (мм) = '); disp(coords); figure()
plot3(coords(1), coords(2),coords(3),'r')
hold on; grid on; title('Конфигурация робота')
x = [0,T(1,1),T(2,1),T(3,1),T(4,1),T(5,1),T(6,1),T(7,1)];
y = [0,T(1,2),T(2,2),T(3,2),T(4,2),T(5,2),T(6,2),T(7,2)];
z = [0,T(1,3),T(2,3),T(3,3),T(4,3),T(5,3),T(6,3),T(7,3)];
for i = 1:8
    if abs(x(i)) < 0.0000000001
        x(i)=0;
    end
    if abs(y(i)) < 0.0000000001
        y(i)=0;
    end
    if abs(z(i)) < 0.0000000001
        z(i)=0;
    end
end
plot3(x,y,z,'o-r','LineWidth',3)
hold off

%% Представление Денавита-Хартенберга
function [A, T, coords] = den_hart(teta, alp, a, d)
A = 1; T = zeros(7,3);
for j = 1:7
    B = [cos(teta(j)) -cos(alp(j))*sin(teta(j)) sin(alp(j))*sin(teta(j))
          a(j)*cos(teta(j));
          sin(teta(j)) cos(alp(j))*cos(teta(j)) -sin(alp(j))*cos(teta(j))
          a(j)*sin(teta(j));
          0 sin(alp(j)) cos(alp(j)) d(j);
          0 0 0 1];
    A = A*B;
    T(j,1) = A(1,4);
    T(j,2) = A(2,4);
    T(j,3) = A(3,4);
end
```

```
coords = [A(1,4),A(2,4), A(3,4)];
end
```

В таблицу 1 представлены данные о результатах решения ПЗК для различных наборов вектора угловых перемещений робота  $\theta_i$ .

Таблица 1 – Результаты решения ПЗК

Обобщенные координаты, °	Декартовы координаты, мм
[0 0 0 0 0]	[890 0 1200]
[0 45 0 0 0]	[1243.6 0 1053.6]
[0 -45 0 0 0]	[536.4 0 1053.6]
[0 0 -25 25 0]	[827.2 0 1483.2]
[0 -45 -25 25 0]	[473.7 0 1336.7]
[0 0 45 -45 0]	[693.7 0 726.2]
[0 45 45 -45 0]	[1047.3 0 579.8]

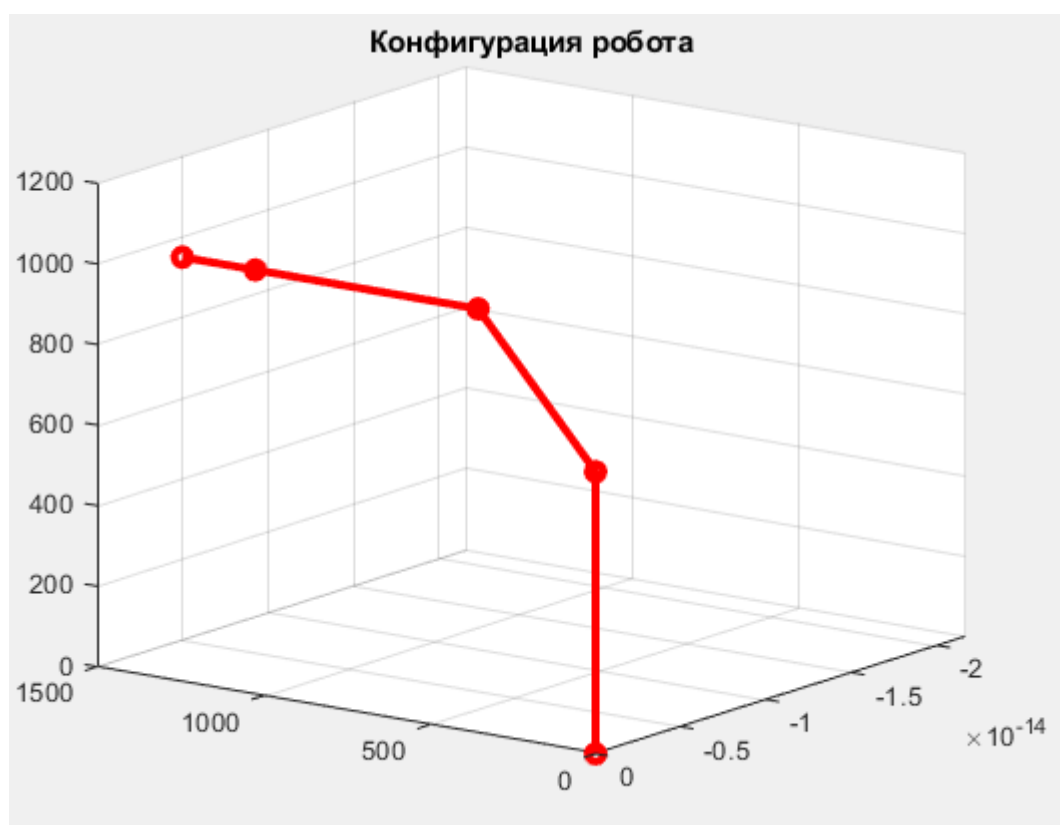


Рисунок 1 – Конфигурация робота при значениях вектора обобщенных координат равное [0 45 0 0 0] при решении ПЗК

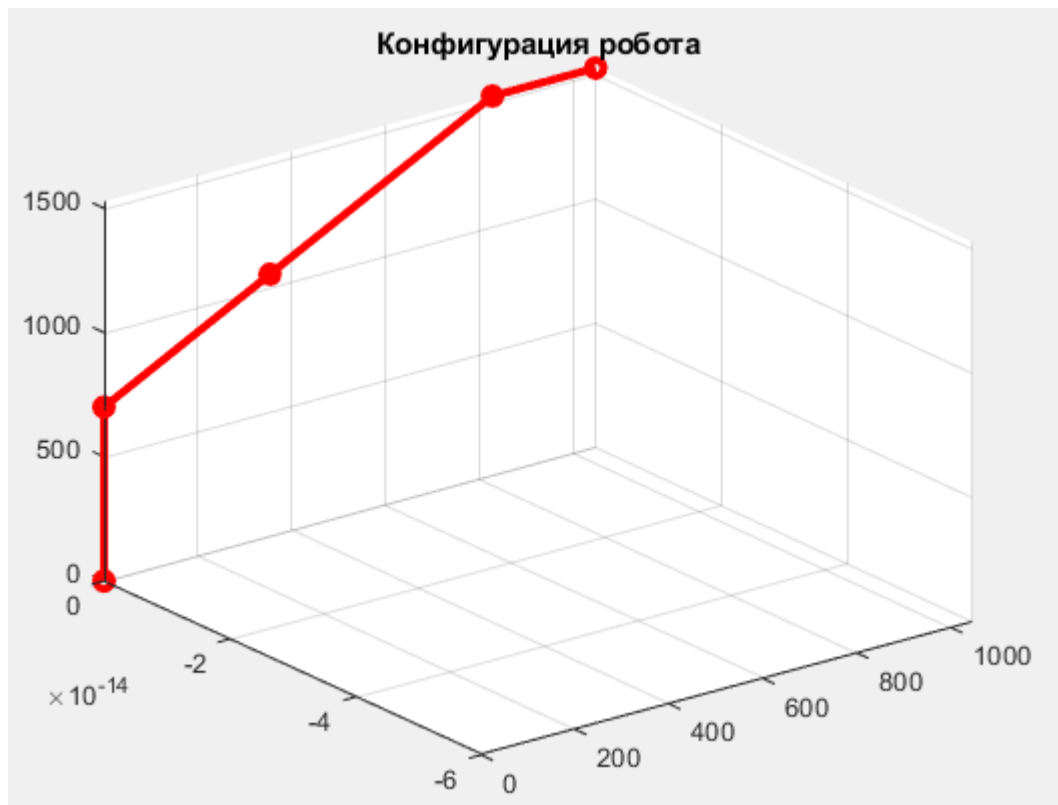


Рисунок 2 – Конфигурация робота при значениях вектора обобщенных координат равное [0 45 45 -45 0] при решении ПЗК

2. Для решения ОЗК было разработано программное обеспечение на основе генетического алгоритма (функция `ga()`) в Matlab для поиска экстремума функции квадрата невязки между заданными координатами и решением ПЗК при помощи генетического алгоритма. Также был применен метод градиентного спуска и симплексный метод, чтобы сравнить эффективность каждого метода. В таблицу 2 сведены данные о результатах решения ОЗК для наборов положения робота, представленных в таблице 1.

### Листинг программы

```
% Решение ОЗК для ТУР10-К
clear all; clc; close all;
%Если использовать ограничения
%lb = [-135 -45 -45 -90 -180];
%ub = [135 45 25 90 180];
%t = ga(@fun,5,[],[],[],[],lb,ub);
%Генетический алгоритм
%t = ga(@func,5);
%Симплексный метод
t = fminsearch(@func,[0 0 0 0 0 0]);
%Градиентный метод
%t = fminunc(@func,[0 0 0 0 0 0]);

PZKTYR10_graphic(t);
disp('Обобщенные координаты (град) = ');
disp(t);

%% Целевая Функция
```

```

function [f] = func (teta)
%x1 = 890; y1 = 0; z1 = 1200;
% x1= 1244; y1 = 0; z1 = 1054;
%x1 = 536; y1 = 0; z1 = 1054;
%x1 = 827; y1 = 0; z1 = 1483;
%x1 = 473; y1 = 0; z1 = 1337;
%x1 = 694; y1 = 0; z1 = 726;
% x1 = 1048; y1 = 0; z1 = 580;
[coords] = PZKTYR10(teta);
x = coords(1); y = coords(2); z = coords(3);
%Целевая функция. Задавается пользователем.
f = sqrt((x1 - x)^2 + (y1 - y)^2 + (z1 - z)^2);
end

```

Таблица 2 – Результаты решения ОЗК

Декартовы координаты, мм	Обобщенные координаты, ° (Генетический алгоритм)	Обобщенные координаты, ° (Симплексный метод)	Обобщенные координаты, ° (Градиентный метод)
[890 0 1200]	[0 0 1 -2 1]	[0 0 0 0.625 0.625]	[0 0 0 0 0]
[1244 0 1054]	[0 45 -1 2 4]	[0 46.2 16.7. -37 -10.1]	[0 45 0.23 0 0 ]
[536 0 1054]	[0 -45 0 -1 97]	[0 -44 -15 6.7 28]	[0 -45 0 0 0]
[827 0 1483]	[0 -1 -15 -13 -108]	[0 -1.67 -16 - 9.2 0]	[0 -1.8 -4 0 0]
[473 0 1337]	[0 -45 -25 25 70]	[0 -46 -10 21.2 28]	[0 -47 -4 0 0]
[694 0 726]	[0 10 22 72 -15]	[0 -7 7 -23 4.8]	[0 -6.8 7.5 0 0]

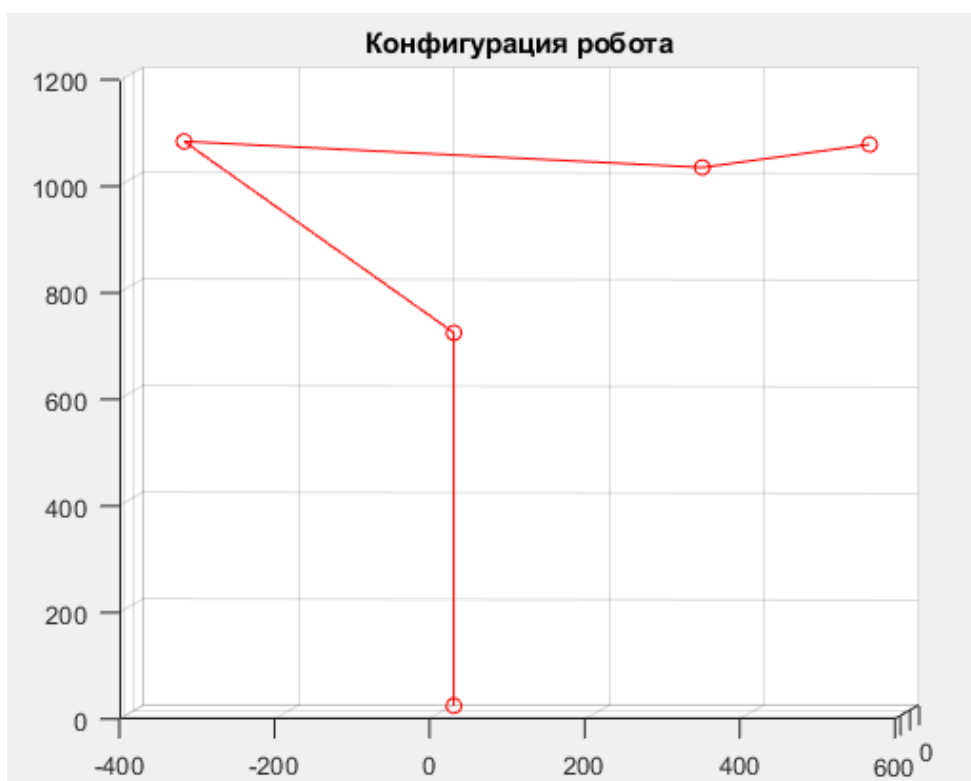


Рисунок 3 – Конфигурация робота при заданных декартовых координатах [536 0 1054] при решении ОЗК симплексным методом

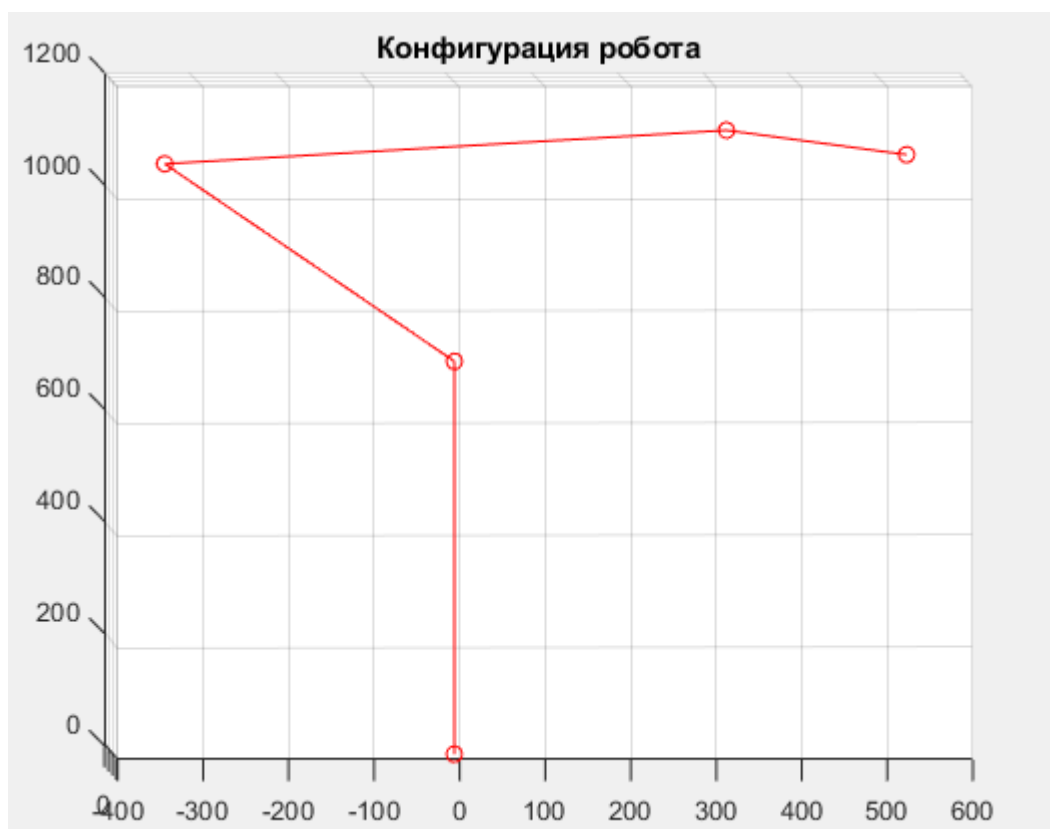


Рисунок 4 – Конфигурация робота при заданных декартовых координатах [536 0 1054] при решении ОЗК генетическим алгоритмом

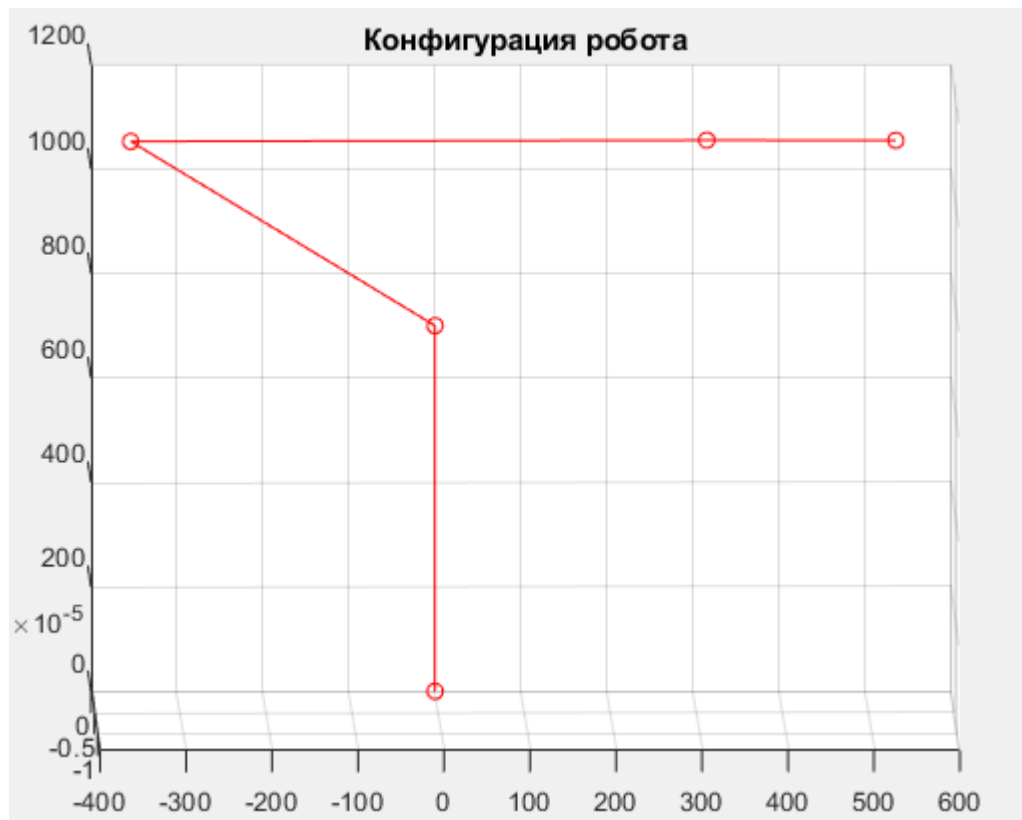


Рисунок 5 – Конфигурация робота при заданных декартовых координатах [536 0 1054] при решении ОЗК градиентным спуском

Как можно увидеть, каждый метод выдает собственную конфигурацию робота с различными обобщенными координатами, поэтому выбор метода оптимизации для ОЗК зависит от конфигурации робота, наличия ограничений на перемещения, требования к времени расчета и точности.

**Вывод:** было разработано программное обеспечение в Matlab для решения прямой (ПЗК) и обратной задачи кинематики (ОЗК) для робота-манипулятора ТУР-10К с учетом особенностей его кинематики.