

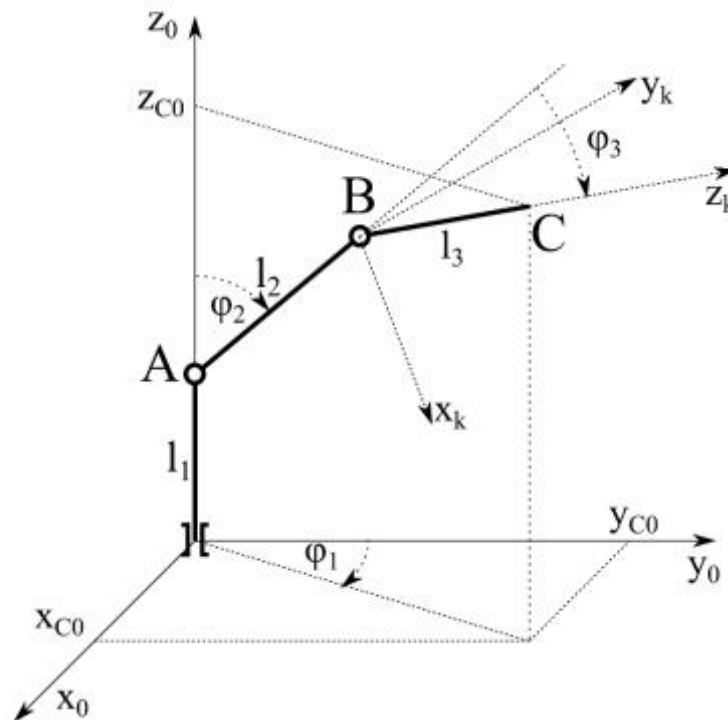
Robotika

## **Zadanie 1**

# **Vizualizácia priamej kinematickej úlohy**

# Vizualizácia priamej kinematickej úlohy

**Zadanie:** Navrhните a realizujte vizualizáciu robotického ramena uvedeného na obr. 1. Na tomto type zadania by ste si mali precvičiť implementáciu homogénnych transformácií a zafixovať tak preberané učivo Priamej kinematickej úlohy trojramenného manipulátora typu RRR. Všetko potrebné bolo odvodené na cvičeniach.



Obr. 1 Robotické rameno typu RRR

## Parametre manipulátora:

$l_1=300$  [mm],  $l_2=300$  [mm],  $l_3=200$  [mm],  $\varphi_1=\langle -160^\circ, 160^\circ \rangle$ ,  $\varphi_2=\langle -50^\circ, 130^\circ \rangle$ ,  $\varphi_3=\langle -30^\circ, 60^\circ \rangle$

V rámci riešenia zadania sa zamerajte na nasledovné úlohy:

1. Vykreslenie manipulátora v 3D priestore v zvolenej konfigurácii (vyskúšajte rôzne hodnoty uhlov  $\varphi$  (obr. 2).
2. Vykreslenie jednotlivých pomocných súradných systémov 0-k (x-červenou farbou, y-zelenou, z-modrou)(obr. 2).
3. Vykreslenie obálky pracovného priestoru daného manipulátora v básovej rovine manipulátora  $X_0Z_0$ , tiež aj v rovine  $X_0Y_0$ .(obr. 3)

## Teoretický úvod

Pre nastavenie manipulátora do požadovanej polohy použijeme transformačné matice, odvodené na cvičeniach

Rotačná matica okolo z-osi

$$R_z(\varphi) = \begin{pmatrix} \cos(\varphi) & -\sin(\varphi) & 0 & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Rotačná matica okolo y-osi

$$R_y(\varphi) = \begin{pmatrix} \cos(\varphi) & 0 & \sin(\varphi) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\varphi) & 0 & \cos(\varphi) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Rotačná matica okolo x-osi

$$R_x(\varphi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\varphi) & -\sin(\varphi) & 0 \\ 0 & \sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Translačná matica pozdĺž x-osi

$$T_x(d) = \begin{pmatrix} 1 & 0 & 0 & d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Translačná matica pozdĺž y-osi

$$T_y(d) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Translačná matica pozdĺž z-osi

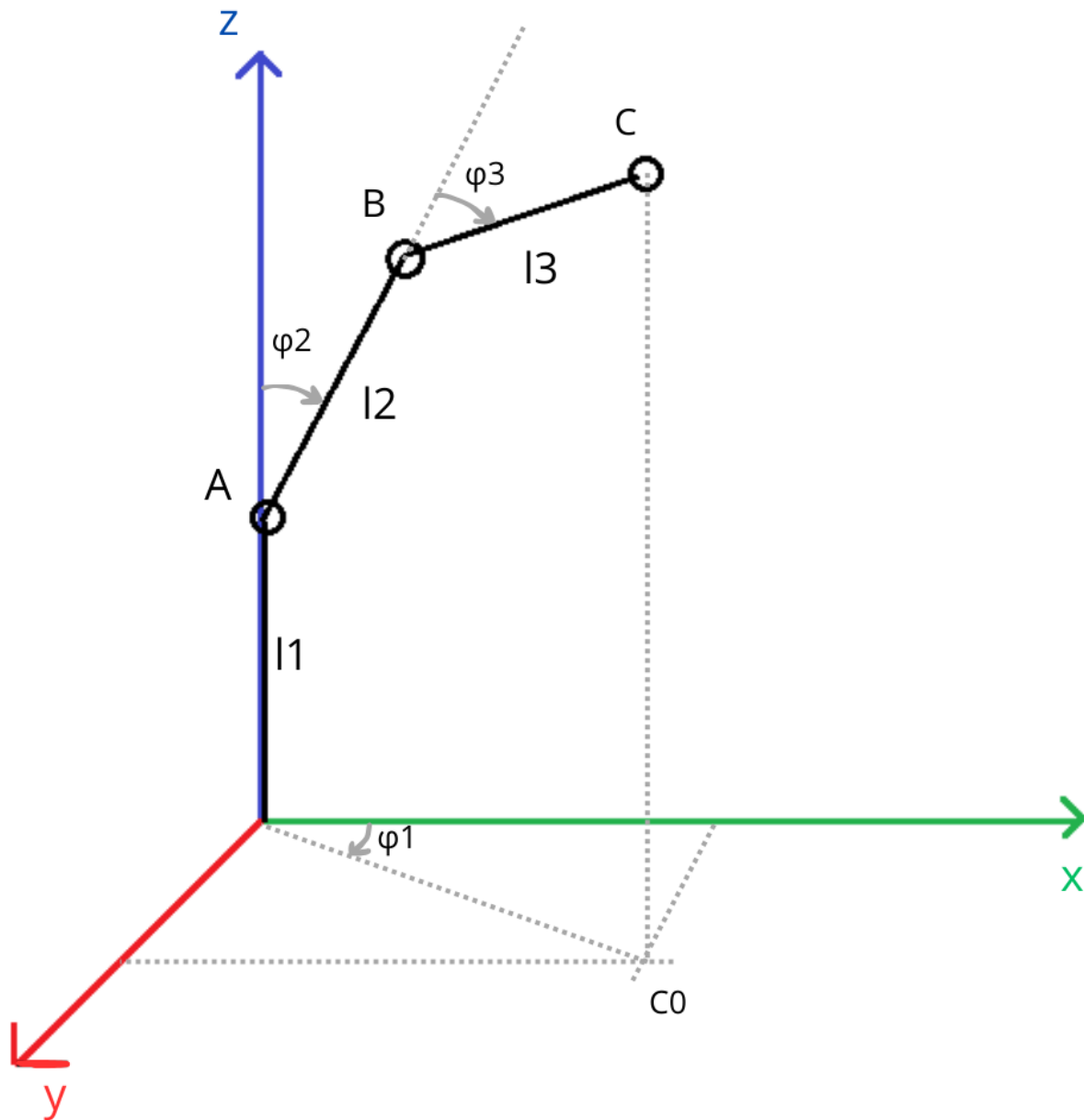
$$T_z(d) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Nulový polohový vektor

$$p_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Každá matica je zodpovedná za pohyb manipulatora v priestore.

Na to aby sme dostali požadovanú polohu každého bodu(kĺbu) manipulatora v priestore, potrebujeme prenásobiť zodpovedajúce matice dosadiac požadované hodnoty uhlov a zadané dĺžky ramien.



Obrázok 1 – náčrt manipulatora

Nech  $\phi_1 = 30^\circ$ ,  $\phi_2 = 60^\circ$ ,  $\phi_3 = 45^\circ$ .  $l_1 = 300[\text{mm}]$ ,  $l_2 = 300[\text{mm}]$ ,  $l_3 = 200[\text{mm}]$  podľa zadania(Obr.1).

Teda bod A =  $R_z(\phi_1) * T_z(l_1) * p_0$

bod B =  $R_z(\pi/2 - \phi_1) * T_z(l_1) * R_y(\phi_2) * T_z(l_2) * p_0$

bod C =  $R_z(\pi/2 - \phi_1) * T_z(l_1) * R_y(\phi_2) * T_z(l_2) * R_y * T_z(l_3) * p_0$  (Obr.1)

## Vykreslenie manipulátora v 3D priestore

Skript na implementáciu riešenia v prostredí Matlab:

```
clc
clear
close all
```

```

x1 = 1;
y1 = 1;
z1 = 1;

l1 = 300;
l2 = 300;
l3 = 200;

phi1 =30;
phi2 =60;
phi3 =45;

quiver3(x1,y1,z1,100,0,0,'r','LineWidth',1);
quiver3(x1,y1,z1,0,100,0,'g','LineWidth',1);
quiver3(x1,y1,z1,0,0,100,'b','LineWidth',1);

v0 = [x1, y1, z1, 1];

A = Rz(phi1) * Tz(l1) * v0';
A = A' ;

plot3([v0(1), A(1)], [v0(2), A(2)], [v0(3), A(3)], 'LineWidth', 2, 'Color', 'b');
hold on;

plot3(v0(1), v0(2), v0(3), 'ro', 'MarkerSize', 6, 'MarkerFaceColor', 'r');
plot3(A(1), A(2), A(3), 'ro', 'MarkerSize', 6, 'MarkerFaceColor', 'r');

quiver3(A(1),A(2),A(3),100,0,0,'r','LineWidth',1);
quiver3(A(1),A(2),A(3),0,100,0,'g','LineWidth',1);
quiver3(A(1),A(2),A(3),0,0,100,'b','LineWidth',1);

B = Rz(phi1) * Tz(l1) * Ry(phi2) *Tz(l2) * v0'
B = B'

plot3([A(1), B(1)], [A(2), B(2)], [A(3), B(3)], 'LineWidth', 2, 'Color', 'b');

plot3(B(1), B(2), B(3), 'ro', 'MarkerSize', 6, 'MarkerFaceColor', 'r');

quiver3(B(1),B(2),B(3),100,0,0,'r','LineWidth',1);
quiver3(B(1),B(2),B(3),0,100,0,'g','LineWidth',1);
quiver3(B(1),B(2),B(3),0,0,100,'b','LineWidth',1);

C = Rz(phi1) * Tz(l1) * Ry(phi2) *Tz(l2)* Ry(phi3) *Tz(l3) * v0'
C = C';

plot3([B(1), C(1)], [B(2), C(2)], [B(3), C(3)], 'LineWidth', 2, 'Color', 'b');

plot3(C(1), C(2), C(3), 'ro', 'MarkerSize', 6, 'MarkerFaceColor', 'r');

quiver3(C(1),C(2),C(3),100,0,0,'r','LineWidth',1);
quiver3(C(1),C(2),C(3),0,100,0,'g','LineWidth',1);
quiver3(C(1),C(2),C(3),0,0,100,'b','LineWidth',1);

axis equal;
axis([-600 600 -600 600 0 600]);
grid on;
xlabel('X');

```

```

ylabel('Y');
zlabel('Z');
function Rotatex = Rx(phi)
    Rotatex = [1 0 0 0; 0 cosd(phi) -sind(phi) 0; 0 sind(phi) cosd(phi) 0; 0 0 0 1];
end

function Rotatey = Ry(phi)
    Rotatey = [cosd(phi) 0 sind(phi) 0; 0 1 0 0; -sind(phi) 0 cosd(phi) 0; 0 0 0 1];
end

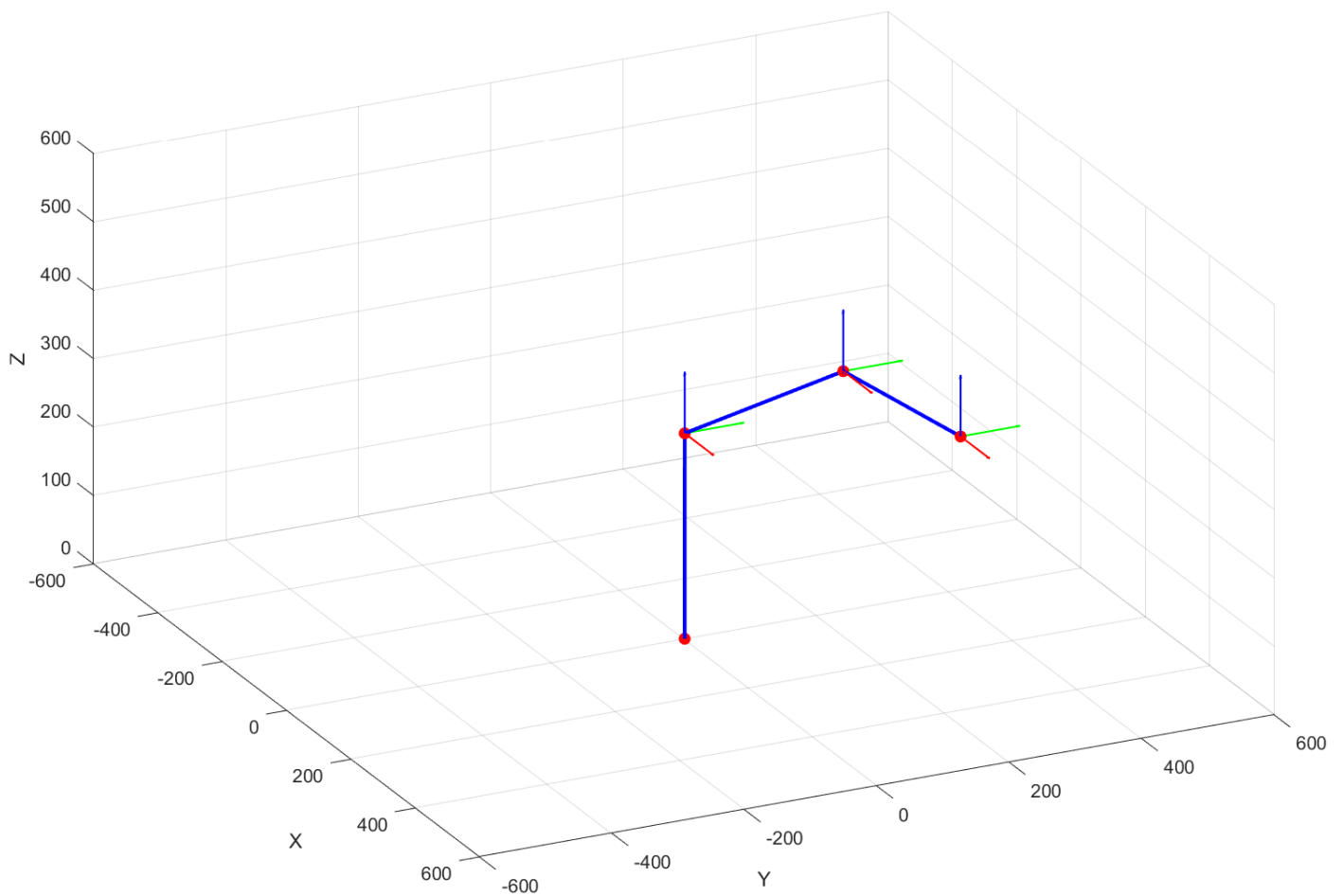
function Rotatez = Rz(phi1)
    Rotatez = [cosd(phi1) -sind(phi1) 0 0; sind(phi1) cosd(phi1) 0 0; 0 0 1 0; 0 0 0 1];
end

function Translatex = Tx(l)
    Translatex = [1 0 0 l; 0 1 0 0; 0 0 1 0; 0 0 0 1];
end

function Translatey = Ty(l)
    Translatey = [1 0 0 0; 0 1 0 l; 0 0 1 0; 0 0 0 1];
end

function Translatez = Tz(l)
    Translatez = [1 0 0 0; 0 1 0 0; 0 0 1 l; 0 0 0 1];
end

```



Obrázok 2 - grafické zobrazenie manipulatora v prostredí Matlab

# Vykreslenie obálky pracovného priestoru daného manipulátora v rovine XZ

Vykreslenie pracovného priestoru navrhne tak, že v cykle prejdeme rozsahmi uhlov  $\phi_2 = \langle -50^\circ, 130^\circ \rangle$  a  $\phi_3 = \langle -30^\circ, 60^\circ \rangle$ , ktoré menia polohu manipulatora v rovine XZ . Každý raz, keď prechádzame cyklom, budeme ratať polohu bodu C v priestore a označovať bodkou.

Skript na implementáciu riešenia v prostredí Matlab:

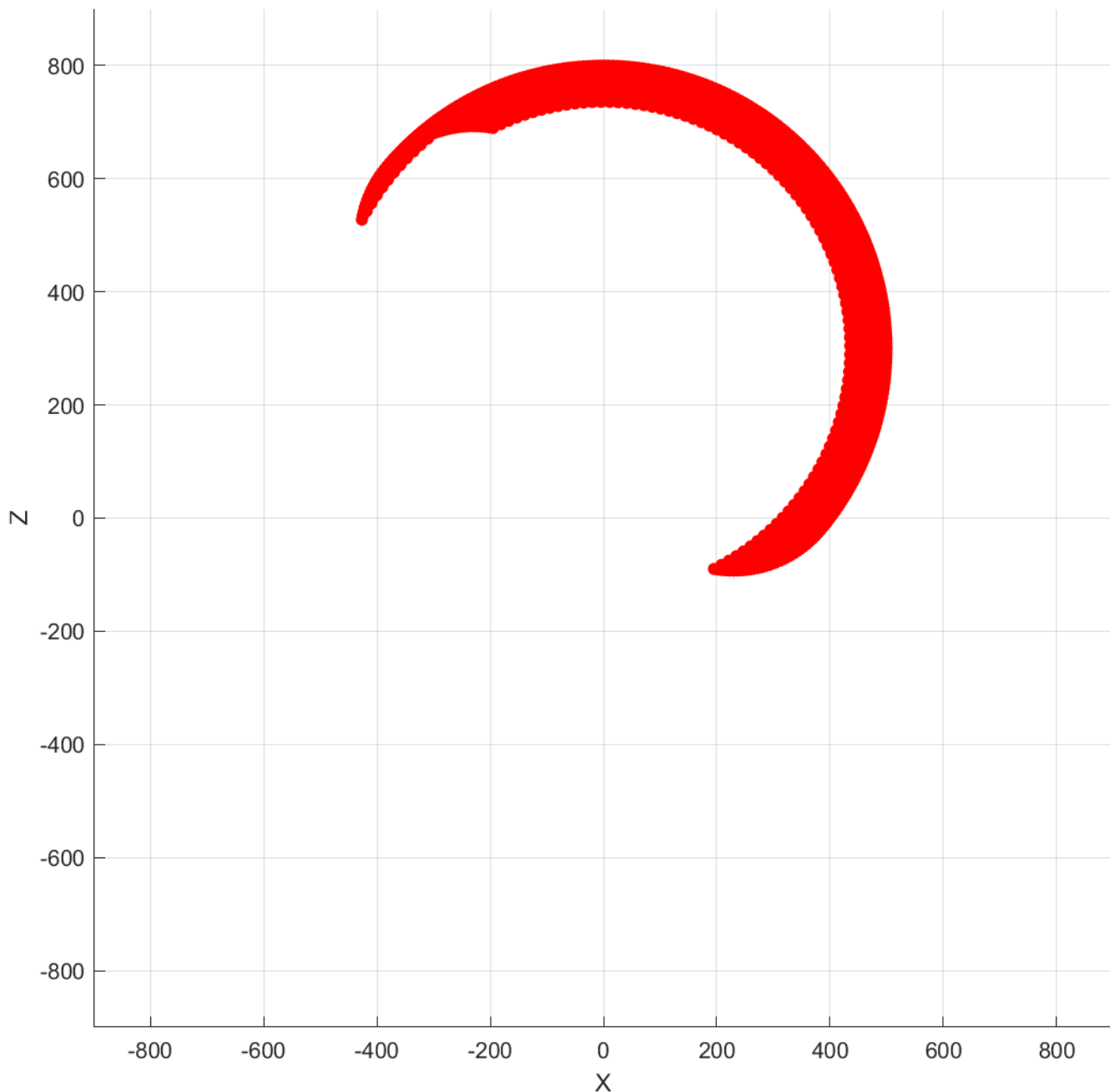
```
%pracovny priestor XZ
```

```
for phi2= -50:2:130
    for phi3= -30:2:60
        C = Rz(0) * Tz(l1) * Ry(phi2) *Tz(l2)* Ry(phi3) *Tz(l3) * v0'
        C = C'

        plot3(C(1), C(2), C(3), '-o', 'Color', 'r', 'MarkerSize', 5, 'MarkerFaceColor',
'r');

    end
end

axis equal;
axis([-900 900 -900 900 -900 900]);
grid on;
xlabel('X');
ylabel('Y');
zlabel('Z');
```



Obrázok 3 - Vykreslenie pracovného priestoru daného manipulátora v rovine XZ

## Vykreslenie obálky pracovného priestoru daného manipulátora v rovine XY

Vykreslenie pracovného priestoru navrhne tak, že v cykle prejdeme rozsahmi všetkých uhlov  $\phi_1 = \langle -160^\circ, 160^\circ \rangle$ ,  $\phi_2 = \langle -50^\circ, 130^\circ \rangle$  a  $\phi_3 = \langle -30^\circ, 60^\circ \rangle$  a vyratame súradnice bodu C vo všetkých možných polohách, ale vykreslíme iba pracovný priestor v rovine XY.

Skript na implementáciu riešenia v prostredí Matlab:

```
figure
hold on;
%pracovny priestor XY
for phi1= -160:2:160
    for phi2= -50:10:130
        for phi3= -30:10:60
            C = Rz(phi1) * Tz(l1) * Ry(phi2) * Tz(l2) * Ry(phi3) * Tz(l3) * v0'
            C = C'

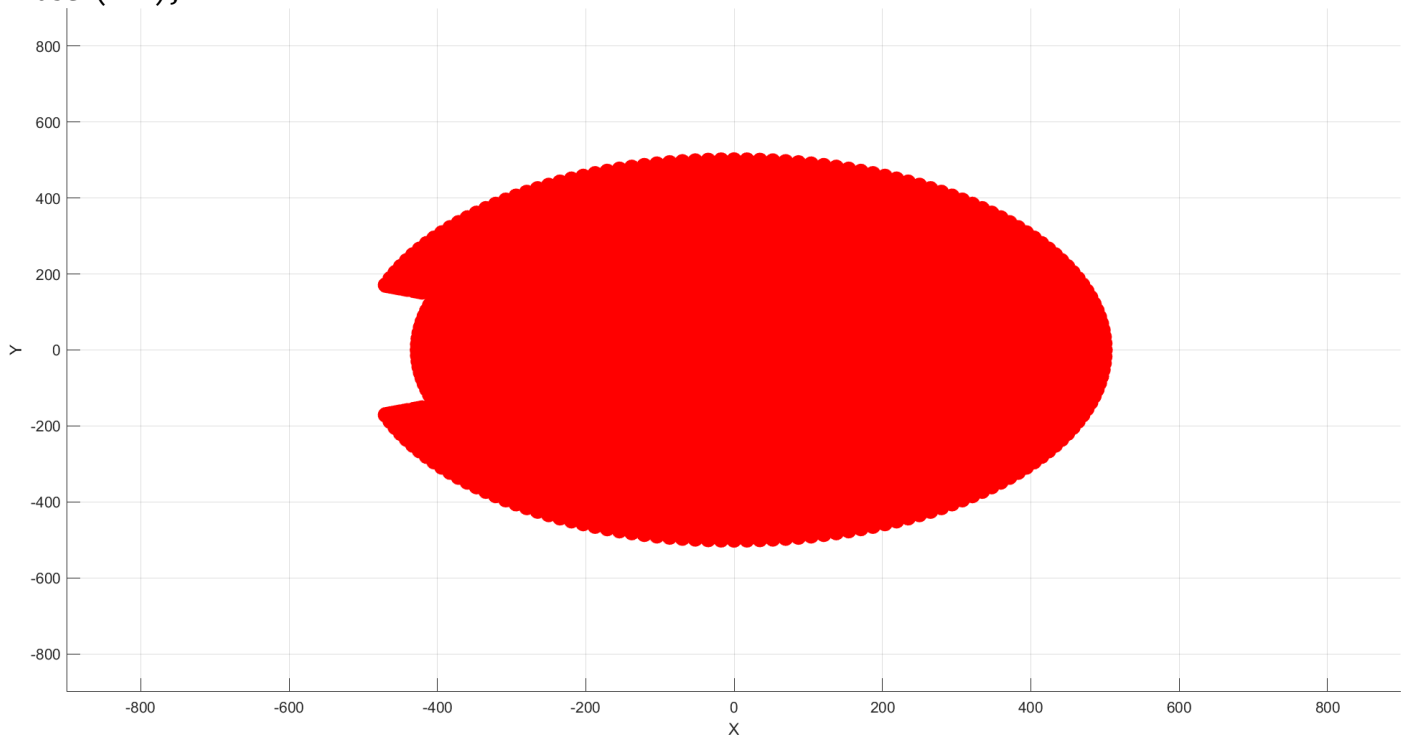
            plot3(C(1), C(2), 0, '-o', 'Color', 'r', 'MarkerSize', 5, 'MarkerFaceColor', 'r');
```



```

end
end
end
xlabel('X');
ylabel('Y');
zlabel('Z');

```



Obrázok 4- Vykreslenie pracovného priestoru daného manipulátora v rovine XY

## Záver

V rámci riešenia zadania som navrhol jednoduchý softvér pre vykreslenie manipulátora v 3D priestore v zvolenej konfigurácii v prostredí Matlab. Okrem toho aplikácia umožňuje aj vykreslenie obálky pracovného priestoru manipulátora v básovej rovine XZ, tiež aj v rovine XY. Pre návrh som používal znalosti a poznámky, ktoré som dostal na cvičeniach z predmetu Robotika.

Zadanie som vypracoval sám. Čestne prehlasujem, že som ho neskopíroval a nikomu inému neposkytol. Nech mi je Isaac Asimov svedkom.

## Celkový kod z Matlabu

Kod na vykreslenie manipulátora v 3D priestore:

```

clc
clear
close all
x1 = 1;
y1 = 1;
z1 = 1;
%dlzky ramien
l1 = 300;
l2 = 300;
l3 = 200;
v0 = [x1, y1, z1, 1];

%hodnoty uhlov
phi1 = 30;
phi2 = 60;
phi3 = 45;

```

```

%vykreslenie jednotlivych pomocnych suradnych systemov
quiver3(x1,y1,z1,100,0,0,'r','LineWidth',1);
quiver3(x1,y1,z1,0,100,0,'g','LineWidth',1);
quiver3(x1,y1,z1,0,0,100,'b','LineWidth',1);

%poloha bodu A
A = (Rz(phi1) * Tz(l1) * v0');
A = A';

%vykreslenie ramena l1 a klbov manipulatora
plot3([v0(1), A(1)], [v0(2), A(2)], [v0(3), A(3)], 'LineWidth', 2, 'Color', 'b');
hold on;

plot3(v0(1), v0(2), v0(3), 'ro', 'MarkerSize', 6, 'MarkerFaceColor', 'r');
plot3(A(1), A(2), A(3), 'ro', 'MarkerSize', 6, 'MarkerFaceColor', 'r');
%vykreslenie jednotlivych pomocnych suradnych systemov pre bod A
quiver3(A(1),A(2),A(3),100,0,0,'r','LineWidth',1);
quiver3(A(1),A(2),A(3),0,100,0,'g','LineWidth',1);
quiver3(A(1),A(2),A(3),0,0,100,'b','LineWidth',1);

%poloha bodu B
B = (Rz(phi1) * Tz(l1) * Ry(phi2) *Tz(l2) * v0')
B = B';
%vykreslenie ramena l2 a klbu manipulatora B
plot3([A(1), B(1)], [A(2), B(2)], [A(3), B(3)], 'LineWidth', 2, 'Color', 'b');

plot3(B(1), B(2), B(3), 'ro', 'MarkerSize', 6, 'MarkerFaceColor', 'r');

%vykreslenie jednotlivych pomocnych suradnych systemov pre bod B
quiver3(B(1),B(2),B(3),100,0,0,'r','LineWidth',1);
quiver3(B(1),B(2),B(3),0,100,0,'g','LineWidth',1);
quiver3(B(1),B(2),B(3),0,0,100,'b','LineWidth',1);

%poloha bodu C
C = (Rz(phi1) * Tz(l1) * Ry(phi2) *Tz(l2)* Ry(phi3) *Tz(l3) * v0')
C = C';
%vykreslenie ramena l3 a klbu manipulatora C
plot3([B(1), C(1)], [B(2), C(2)], [B(3), C(3)], 'LineWidth', 2, 'Color', 'b');

plot3(C(1), C(2), C(3), 'ro', 'MarkerSize', 6, 'MarkerFaceColor', 'r');

%vykreslenie jednotlivych pomocnych suradnych systemov pre bod C
quiver3(C(1),C(2),C(3),100,0,0,'r','LineWidth',1);
quiver3(C(1),C(2),C(3),0,100,0,'g','LineWidth',1);
quiver3(C(1),C(2),C(3),0,0,100,'b','LineWidth',1);

%nastavenie mriezky
axis equal;
axis([-600 600 -600 600 0 600]);
grid on;
xlabel('X');
ylabel('Y');
zlabel('Z');
%rotacne a translacne matice
function Rotatex = Rx(phi)
    Rotatex = [1 0 0 0; 0 cosd(phi) -sind(phi) 0; 0 sind(phi) cosd(phi) 0; 0 0 0 1];
end

function Rotatey = Ry(phi)
    Rotatey = [cosd(phi) 0 sind(phi) 0; 0 1 0 0; -sind(phi) 0 cosd(phi) 0; 0 0 0 1];

```

```

end

function Rotatez = Rz(phi1)
    Rotatez = [cosd(phi1) -sind(phi1) 0 0; sind(phi1) cosd(phi1) 0 0; 0 0 1 0; 0 0 0 1];
end

function Translatex = Tx(l)
    Translatex = [1 0 0 l; 0 1 0 0; 0 0 1 0; 0 0 0 1];
end

function Translatey = Ty(l)
    Translatey = [1 0 0 0; 0 1 0 l; 0 0 1 0; 0 0 0 1];
end

function Translatez = Tz(l)
    Translatez = [1 0 0 0; 0 1 0 0; 0 0 1 l; 0 0 0 1];
end

```

Kod na vykreslenie obálky pracovného priestoru daného manipulátora v rovinách XZ a XY:

```

clc;
clear all;

x0 = 0;
y0 = 0;
z0 = 0;
%dlzky ramien
l1 = 300;
l2 = 300;
l3 = 200;
%nulovy polohovy vektor
v0 = [x0, y0, z0, 1];

figure
hold on;
%pracovny priestor XY
for phi1= -160:2:160
    for phi2= -50:10:130
        for phi3= -30:10:60
            C = Rz(phi1) * Tz(l1) * Ry(phi2) *Tz(l2)* Ry(phi3) *Tz(l3) * v0'
            C = C'
            plot3(C(1), C(2), 0, '-o', 'Color', 'r', 'MarkerSize', 10, 'MarkerFaceColor', 'r');
        end
    end
end
%nastavenie mriezky
axis([-900 900 -900 900 -900 900]);
xlabel('X');
ylabel('Y');
zlabel('Z');
grid on;
figure
hold on
%pracovny priestor XZ
    for phi2= -50:2:130
        for phi3= -30:2:60
            C = Rz(0) * Tz(l1) * Ry(phi2) *Tz(l2)* Ry(phi3) *Tz(l3) * v0'
            C = C'
            plot3(C(1), C(2), C(3), '-o', 'Color', 'r', 'MarkerSize', 5, 'MarkerFaceColor',
'r');
        end
    end
end

```

```

%nastavenie mriezky
axis equal;
axis([-900 900 -900 900 -900 900]);
grid on;
xlabel('X');
ylabel('Y');
zlabel('Z');
%rotacne a translacne matice
function Rot_X = Rx(phi)

    Rot_X = [1 0 0 0; 0 cosd(phi) -sind(phi) 0; 0 sind(phi) cosd(phi) 0; 0 0 0 1];
end

function Rot_Y = Ry(phi)

    Rot_Y = [cosd(phi) 0 sind(phi) 0; 0 1 0 0; -sind(phi) 0 cosd(phi) 0; 0 0 0 1];
end

function Rot_Z = Rz(phi1)

    Rot_Z = [cosd(phi1) -sind(phi1) 0 0; sind(phi1) cosd(phi1) 0 0; 0 0 1 0; 0 0 0 1];
end

function Translatex = Tx(l)

    Translatex = [1 0 0 l; 0 1 0 0; 0 0 1 0; 0 0 0 1];
end

function Translatey = Ty(l)

    Translatey = [1 0 0 0; 0 1 0 l; 0 0 1 0; 0 0 0 1];
end

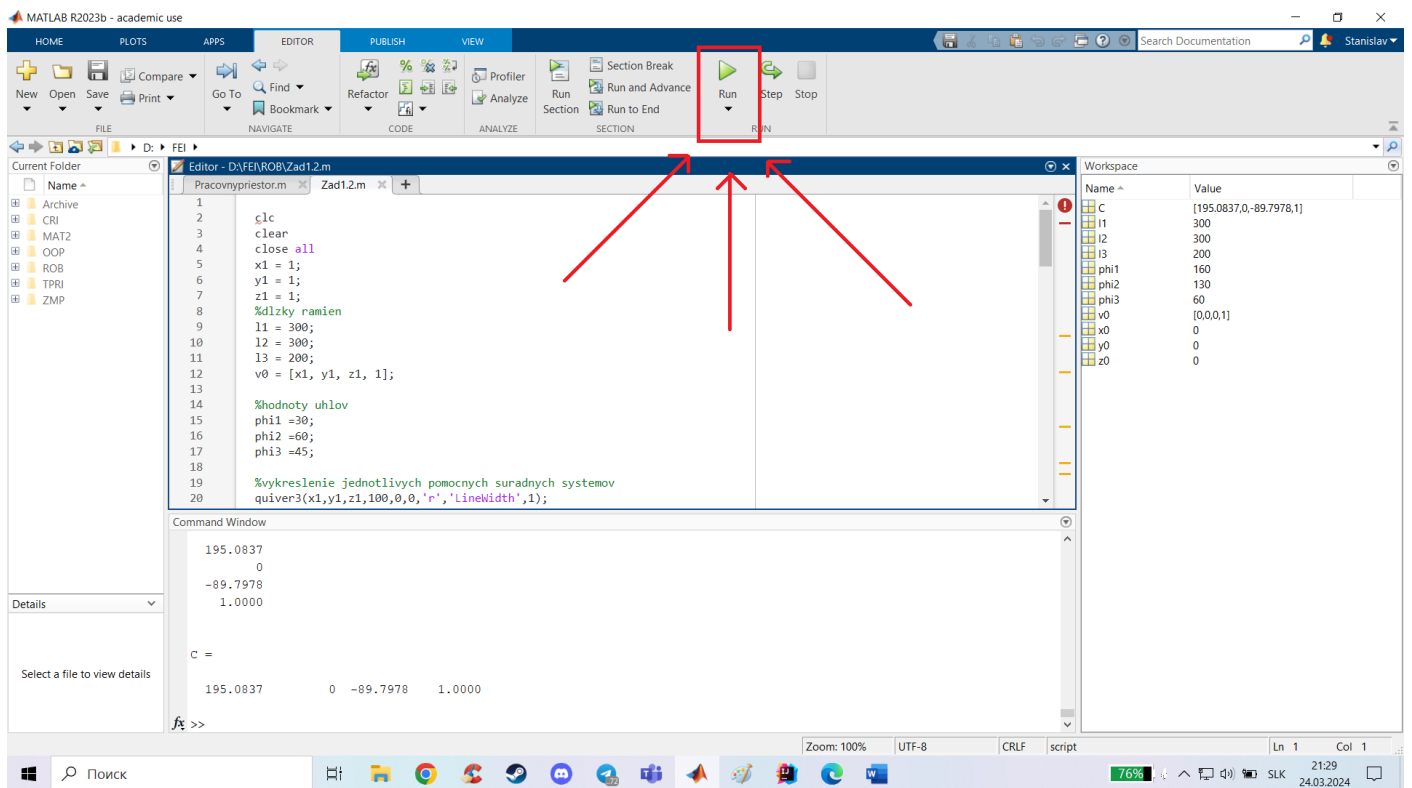
function Translatez = Tz(l)

    Translatez = [1 0 0 0; 0 1 0 0; 0 0 1 l; 0 0 0 1];
end

```

Jednoduchý návod na kompiláciu a spustenie kódu:

Stlačiť tlačidlo Run (Obr.5)



Obrázok 5 - návod na kompiláciu a spustenie kódu