FEI STU ÚRK

Zadanie 1

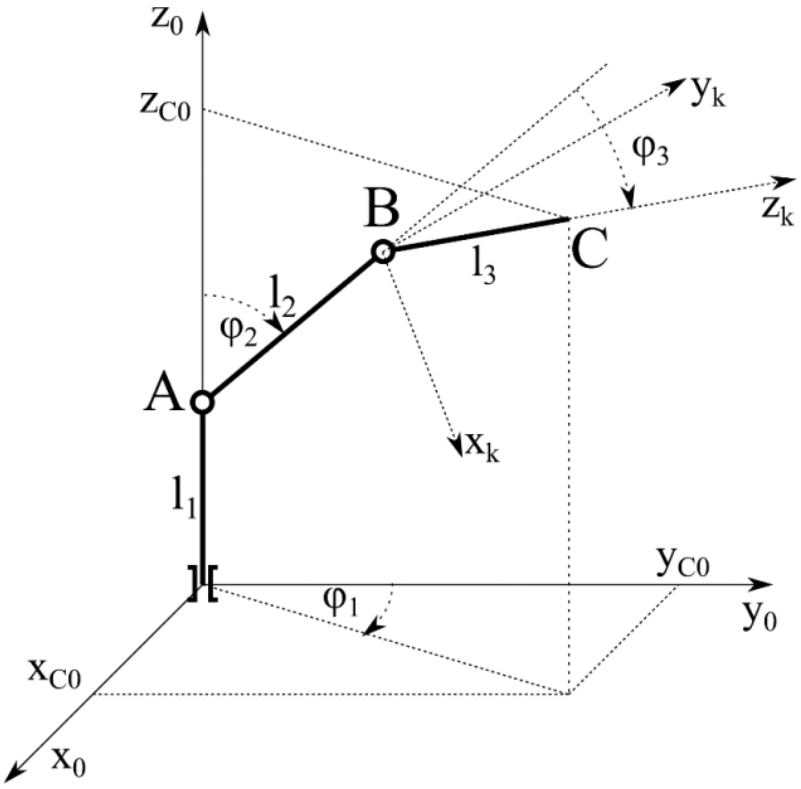
Vizualizácia priamej kinematickej úlohy

Daniel Pacalaj

1.4.2022

ZADANIE

Navrhujeme a realizujeme vizualizáciu robotického ramena uvedeného na obr. 1. Použitím homogénnych transformácií riešime Priamu kinematickú úlohy trojramenného manipulátora typu RRR. Všetko potrebné bolo odvodené na cvičeniach.



Obr. 1

Parametre manipulátora:

l_1=203[mm], l_2=178[mm], l_3=178[mm], ϕ_1=<-90°, 90°>, ϕ_2=<-55°, 125°>, ϕ_3=<0°,150°>

ROZBOR ÚLOHY

Začínajúc v bode [0;0;0], v presnom poradí aplikujeme rotačné a translačné matice, ktoré sme odvodzovali na cvičeniach.

Rotačná matica pre os X
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) & 0 \\ 0 & \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotačná matica pre os Y
$$\begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\phi) & 0 & \cos(\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotačná matica pre os Z $\begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 & 0 \\ \sin(\phi) & \cos(\phi) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Translačná matica pre os X $\begin{bmatrix} 1 & 0 & 0 & d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Translačná matica pre os Y $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Translačná matica pre os Z $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Pre numerické riešenie potrebujeme vynásobiť vhodné matice v správnom poradí. Vychádzajúc z obrázka 1, začneme rotáciou v osi Z, nasleduje translácia v tej istej osi. Tak sa dostávame z počiatku súradnicovej sústavy do bodu A. Násobíme rotačnou maticou v osi Y o uhol ϕ_2 a aplikujeme transláciu v osi Z. Dosiahneme bod B, kde opakujeme rotáciu v osi Y a Z transláciu. Tak sa dostaneme do bodu C, čo je koncový bod nášho ramena.

VIZUALIZÁCIA

Pre simuláciu numerického výpočtu budeme používať prostredie MATLAB, kvôli natívne jednoduchej práci s maticami a pomerne kvalitnej dokumentácií. Pre jednoduchosť aj pre možnosť plne všeobecného použitia použijeme funkcie pre ktoré bude parametrom uhol a návratovou hodnotou je pole polí/ matica.

```
function R = xRotation(fi)

    R = [1 0 0 0;
         0 cosd(fi) -sind(fi) 0;
         0 sind(fi) cosd(fi) 0;
         0 0 0 1];
end
```

Podobné funkcie aplikujeme aj pre transláciu a následne môžeme už konkrétne matice násobiť.

```
A = zRotation(fi1) * zTranslation(l1)*[0;0;0;1];
B = zRotation(fi1) * zTranslation(l1) * yRotation(fi2) * zTranslation(l2)*[0;0;0;1];
C = zRotation(fi1) * zTranslation(l1)* yRotation(fi2) * zTranslation(l2) * yRotation(fi3) * zTranslation(l3)*[0;0;0;1];
```

Následne už použijeme funkciu plot3 kde sú vstupnými parametrami X,Y a Z súradnice bodov A,B a C.

```
plot3(X,Y,Z,'k','LineWidth',3)
```

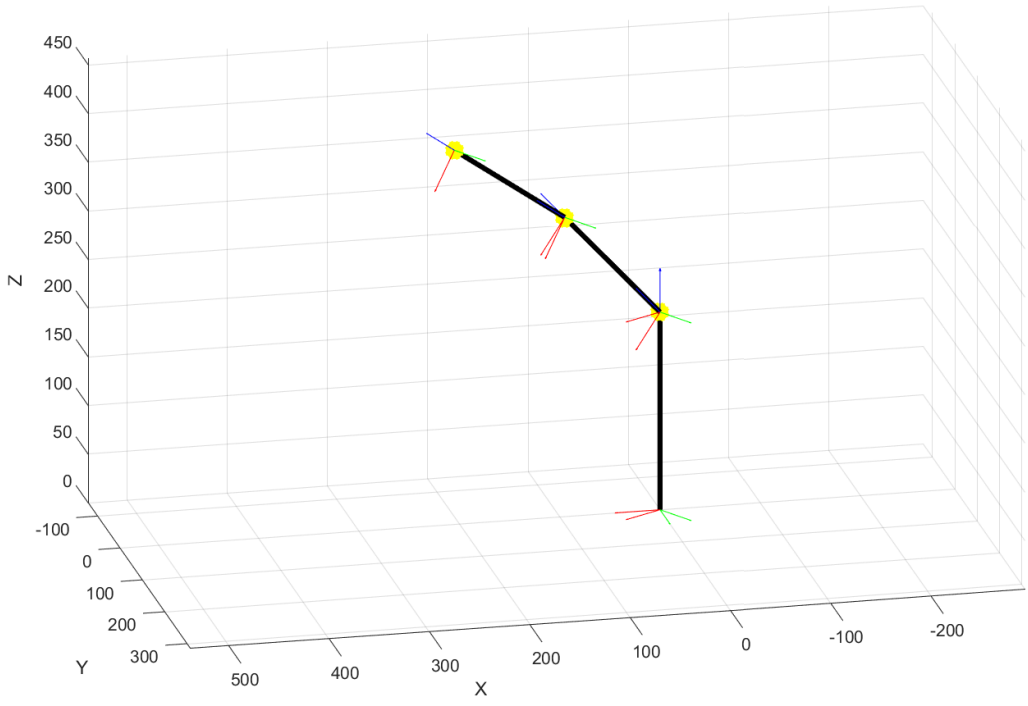
Pre vytlačenie osí nových súradnicových sústav použijeme funkciu quiver3, kde výstupom je vektor s relatívnou veľkosťou. Preto postupne vytvárame osi $X,Y,Z_{1\text{-}6}$.

```
x5 = zRotation(fi1) * yRotation(fi2) * yRotation(fi3) * x0;
y5 = zRotation(fi1) * yRotation(fi2) * yRotation(fi3) * y0;
z5 = zRotation(fi1) * yRotation(fi2) * yRotation(fi3) * z0;
```

A následne to vytlačíme do rovnakého grafu, ako ramená robota.

```
quiver3(B(1),B(2),B(3),y5(1),y5(2),y5(3),'g');
quiver3(B(1),B(2),B(3),x5(1),x5(2),x5(3),'r');
quiver3(B(1),B(2),B(3),z5(1),z5(2),z5(3),'b');
```

Dostaneme už kompletnú vizualizáciu numerického riešenia kinematickej úlohy (obr. 2).

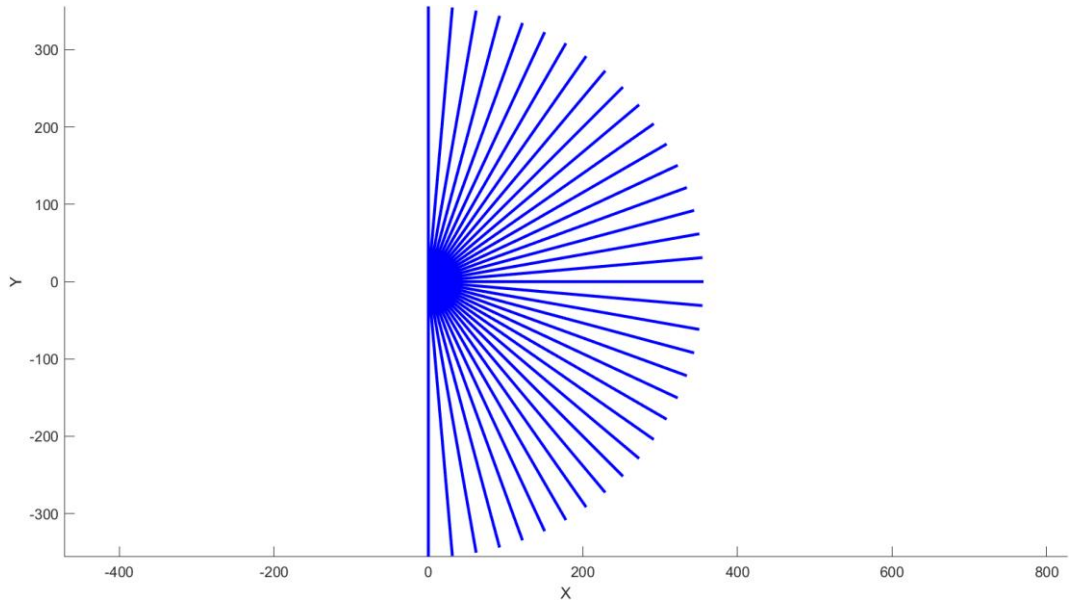


Obr. 2

Ďalšou úlohou je vykreslenie pracovného priestoru v X rovine a Y rovine. Použijeme teda cyklus for kde pre rez v rovine Z budeme inkrementovať uhol ϕ_1 s krokom 5°.

```
for i=-90:5:90
fi1 = i;
```

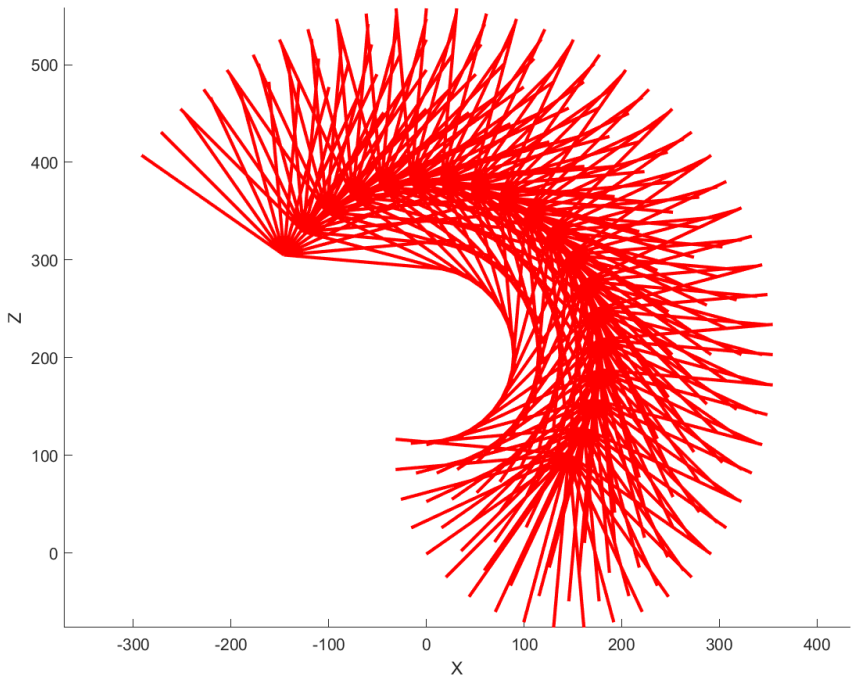
Následným vytlačením dostaneme pracovný priestor v rovine X_0.



Pre rez v rovine Y použijeme vnorený dvojitý cyklus for pre inkrementáciu uhlov ϕ_2 a ϕ_3.

```
for i=-55:10:125
    for j = 0:10:150
fi1 = 0;
fi2 = i;
fi3 = j;
```

Výsledkom je približný pracovný priestor v rovine Y_0.



KOMPILÁCIA A ZDROJOVÝ KÓD

Pre kompiláciu je potrebné iba prostredie MATLAB. Nie je potrebné inštalovať žiadne dodatočné toolboxy.

```
% STU Bratislava 2022
% Daniel Pacalaj
% Robotika LS 2021/2022

clear all
l1 = 203;
l2 = 178;
l3 = 178;

fi1 = 30;
fi2 = 45;
fi3 = 10;

A = zRotation(fi1) * zTranslation(l1)*[0;0;0;1];
B = zRotation(fi1) * zTranslation(l1) * yRotation(fi2) *
zTranslation(l2)*[0;0;0;1];
C = zRotation(fi1) * zTranslation(l1)* yRotation(fi2) *
zTranslation(l2) * yRotation(fi3) *
zTranslation(l3)*[0;0;0;1];
X = [0;A(1);B(1);C(1)];
Y = [0;A(2);B(2);C(2)];
Z = [0;A(3);B(3);C(3)];

%% Jednoduche vykreslenie
y0 = [0;50;0;1];
x0 = [50;0;0;1];
z0 = [0;0;50;1];

x1 = zRotation(fi1) * x0;
y1 = zRotation(fi1) * y0;
z1 = zRotation(fi1) * z0;

x2 = x1 + zTranslation(l1) * [0;0;0;1];
y2 = y1 + zTranslation(l1) * [0;0;0;1];
z2 = z1 + zTranslation(l1) * [0;0;0;1];

x3 = zRotation(fi1) * yRotation(fi2) * x0;
y3 = zRotation(fi1) * yRotation(fi2) * y0;
z3 = zRotation(fi1) * yRotation(fi2) * z0;

x4 = x3 + zTranslation(l2) * [0;0;0;0];
y4 = y3 + zTranslation(l2) * [0;0;0;0];
z4 = z3 + zTranslation(l2) * [0;0;0;0];

x5 = zRotation(fi1) * yRotation(fi2) * yRotation(fi3) * x0;
y5 = zRotation(fi1) * yRotation(fi2) * yRotation(fi3) * y0;
z5 = zRotation(fi1) * yRotation(fi2) * yRotation(fi3) * z0;
```

```
x6 = x5 + zTranslation(l3) * [0;0;0;0];
y6 = y5 + zTranslation(l3) * [0;0;0;0];
z6 = z5 + zTranslation(l3) * [0;0;0;0];


figure(1)
hold on
grid on
axis equal

plot3(X,Y,Z,'k','LineWidth',3)

%Vykreslenie suradnic Xk,Yk a Zk
quiver3(0,0,0,y0(1),y0(2),y0(3),'g');
quiver3(0,0,0,x0(1),x0(2),x0(3),'r');

quiver3(0,0,0,y1(1),y1(2),y1(3),'g');
quiver3(0,0,0,x1(1),x1(2),x1(3),'r');

quiver3(A(1),A(2),A(3),y2(1),y2(2),y2(3),'g');
quiver3(A(1),A(2),A(3),x2(1),x2(2),x2(3),'r');
quiver3(A(1),A(2),A(3),z2(1),z2(2),z2(3),'b');

quiver3(A(1),A(2),A(3),y3(1),y3(2),y3(3),'g');
quiver3(A(1),A(2),A(3),x3(1),x3(2),x3(3),'r');
quiver3(A(1),A(2),A(3),z3(1),z3(2),z3(3),'b');

quiver3(B(1),B(2),B(3),y4(1),y4(2),y4(3),'g');
quiver3(B(1),B(2),B(3),x4(1),x4(2),x4(3),'r');
quiver3(B(1),B(2),B(3),z4(1),z4(2),z4(3),'b');

quiver3(B(1),B(2),B(3),y5(1),y5(2),y5(3),'g');
quiver3(B(1),B(2),B(3),x5(1),x5(2),x5(3),'r');
quiver3(B(1),B(2),B(3),z5(1),z5(2),z5(3),'b');

quiver3(C(1),C(2),C(3),y6(1),y6(2),y6(3),'g');
quiver3(C(1),C(2),C(3),x6(1),x6(2),x6(3),'r');
quiver3(C(1),C(2),C(3),z6(1),z6(2),z6(3),'b');

plot3(A(1),A(2),A(3),'y*','LineWidth',10);
plot3(B(1),B(2),B(3),'y*','LineWidth',10);
plot3(C(1),C(2),C(3),'y*','LineWidth',10);
xlabel("X");
ylabel("Y");
zlabel("Z");
hold off

%% Priemet v X-rovine
figure(2)
hold on
```

```
axis equal
for i=-90:5:90
fi1 = i;
fi2 = 90;
fi3 = 0;
A = zRotation(fi1) * zTranslation(l1)*[0;0;0;1];
B = zRotation(fi1) * zTranslation(l1) * yRotation(fi2) *
zTranslation(l2)*[0;0;0;1];
C = zRotation(fi1) * zTranslation(l1)* yRotation(fi2) *
zTranslation(l2) * yRotation(fi3) *
zTranslation(l3)*[0;0;0;1];
X = [0;A(1);B(1);C(1)];
Y = [0;A(2);B(2);C(2)];

plot(X,Y,'b','LineWidth',2);
end
xlabel("X");
ylabel("Y");

%% Priemet v Y-rovine
figure(3)
hold on
axis equal
for i=-55:10:125
    for j = 0:10:150
fi1 = 0;
fi2 = i;
fi3 = j;
B = zRotation(fi1) * zTranslation(l1) * yRotation(fi2) *
zTranslation(l2)*[0;0;0;1];
C = zRotation(fi1) * zTranslation(l1)* yRotation(fi2) *
zTranslation(l2) * yRotation(fi3) *
zTranslation(l3)*[0;0;0;1];
X = [B(1);C(1)];
Z = [B(3);C(3)];

plot(X,Z,'r','LineWidth',2);
    end
end
xlabel("X");
ylabel("Z");


function R = zRotation(fi)
    R = [cosd(fi) -sind(fi) 0 0;
        sind(fi) cosd(fi) 0 0;
        0 0 1 0;
        0 0 0 1];
end

function R = xRotation(fi)
```

```
    R = [1 0 0 0;
         0 cosd(fi) -sind(fi) 0;
         0 sind(fi) cosd(fi) 0;
         0 0 0 1];
end

function R = yRotation(fi)
    R = [cosd(fi) 0 sind(fi) 0;
         0 1 0 0;
         -sind(fi) 0 cosd(fi) 0;
         0 0 0 1];
end

function T = xTranslation(d)
    T = [1 0 0 d;
         0 1 0 0;
         0 0 1 0;
         0 0 0 1];
end

function T = yTranslation(d)
    T = [1 0 0 0;
         0 1 0 d;
         0 0 1 0;
         0 0 0 1];
end

function T = zTranslation(d)
    T = [1 0 0 0;
         0 1 0 0;
         0 0 1 d;
         0 0 0 1];
End
```

ZDROJE

https://is.stuba.sk/auth/dok_server/slozka.pl?id=236697;download=225011;lang=sk
Dokumentácia k MATLABU
https://uk.mathworks.com/

https://github.com/danielpacalaj/Vizualizacia_priamej_kinematickej_-ulohy.git