Univerza v Ljubljani
Fakulteta za elektrotehniko in računalništvo
Tržaška 25
61000 Ljubljana

Poročilo z Laboratorijskih vaj pri predmetu Robotika 1 Merjenje hitrosti na robotu ADEPT

Simon MALI

22. februar 1996

1 Naloga

Napišite program za gibanje robota po

1. trikotni trajektoriji,
2. krožni trajektoriji.

Med gibanjem robota istočasno merite z robotom tudi koordinate vrha manipolatorja. Prenesite podatke v Matlab in določite hitrost robota kot prvo in pospešek kot drugo diferenco. Trikotno in krožno gibanje ponovite pri različnih:

a) programskih hitrostih (`SPEED=[30 60 90]`),

b) gibanju `MOVE` in

c) `MOVES` ter pri različnih

d) pospeških in pojemkih `ACCEL[30;30 60;60 90;90]` in ugotovite kakšni so profili hitrosti dejansko.

Poročilo:

- Trajektorije gibanja robota
  1. v XY ravnini(x,y),
  2. v odvisnisti od časa x(t),y(t)
- izrišite hitrosti in pospešek
- ADEPT in Matlab program

2 Program (VAL) za merjenje pozicije robota v odvisnosti od časa za trikotnik in krog

Ker nam robot omogoča merjenje položaja, ki ga je izvršil po točno določenem času smo to izkoristili za merjenje hitrosti in pospeškov.

Gibanje po trikotniku

```
.PROGRAM trikot(a)
        MOVES a
        TIMER 1 = 0
        DO
            IF TIMER(1) > 0.015 THEN
                t[i] = TIMER(1)
                HERE k[i]
                TIMER 1 = 0
                i = i+1
            END
        UNTIL STATE(2) == 2
.END
.PROGRAM trikotnik()
        ACCEL 30, 30
        SPEED 100 ALWAYS
        i = 1
        MOVE t3
        CALL trikot(t1)
        CALL trikot(t2)
        CALL trikot(t3)
.END
```

Gibanje po krožnici

```
.PROGRAM krog()
        ACCEL 60, 60
        SPEED 100 ALWAYS
        i = 1
        x0 = 520
        y0 = 120
        z = 800
        r = 100
        FOR kot = 0 TO 360 STEP 5
            x = r*COS(kot)+x0
            y = r*SIN(kot)+y0
            MOVES TRANS(x, y, z, 0, 180, 0)
            TIMER 1 = 0
            DO
                IF TIMER(1) > 0.015 THEN
                    t[i] = TIMER(1)
                    HERE k[i]
                    TIMER 1 = 0
                    i = i+1
                END
            UNTIL STATE(2) == 2
        END
.END
```

3 Program v jeziku Matlab

Za analizo smo uporabili programski jezik vgrajen v paket Matlab. Posamezne funkcije so lahko uporabniške tako, da jeh napišemo v .m datoteke.

Glavni program za izračun karakteristik (ata.m)

```
function [xfilt,yfilt,vxfilt,vyfilt, ...
        vfilt,axfilt,ayfilt,afilt,t]=ata(ime,tdif);

        [x,y]=kompo(ime);
        [xfilt,yfilt]=myfilt(x,y);

        [t]=cas(xfilt);

        [vx,vy,v]=hitr(x,y,tdif);
        [vxfilt,vyfilt]=myfilt(vx,vy);
        vfilt=sqrt(vxfilt.*vxfilt+vyfilt.*vyfilt);

        [ax,ay,a]=posp1(x,y,tdif);
        [axfilt,ayfilt]=myfilt(ax,ay);
        afilt=sqrt(axfilt.*axfilt+ayfilt.*ayfilt);
end
```

Ker imamo premalo točk bi to pri dvakratnem odvajanju pomenilo, da dobimo preveč šuma zato uporabimo numerični filter, ki je napisan kot program v Matlabu.

```
function [x,y]=myfilt(xvh, yvh)
  [b,a] = cheby(2, 0.5, 15/62.5)
  x = filtfilt(b, a, xvh)
  y = filtfilt(b, a, yvh)
end
```

Hitrosti in pospeške se enostavno izračuna z numeričnim odvajenjem (funkcija diff()).

hitr.m

```
function [vx,vy,v]=hitrost(x,y,tdif)
        xdif=diff(x);
        ydif=diff(y);
        vx=xdif/tdif;
        vy=ydif/tdif;

        vx=[0;vx];
        vy=[0;vy];
        v=sqrt(vx.*vx+vy.*vy);
end
```

posp1.m

```
function [ax,ay,a]=posp1(x,y,tdif)
        [vx,vy,v]=hitr(x,y,tdif)
        xdif=diff(vx);
```

```
        ydif=diff(vy);
        ax=xdif/tdif;
        ay=ydif/tdif;

        ax=[0;ax];
        ay=[0;ay];
        a=sqrt(ax.*ax+ay.*ay);
end
```

Komponente matrike ime

```
function [x,y]=kompo(m)
   x=m(:,1);
   y=m(:,2);
end
```

Funkcije za izris

```
% rise Pozicije, ki so sfiltrirane zaradi boljse preglednosti
function risia(ime,tdif);
  clc;
  [xfilt,yfilt,vxfilt,vyfilt,vfilt,axfilt,ayfilt,afilt,t]=ata(ime,tdif);
  subplot(221), plot(xfilt,yfilt), title('Pozicija')
  subplot(223), plot(t,xfilt), title('Pozicija x(t)')
  subplot(224), plot(t,yfilt), title('Pozicija y(t)')
end

% rise Hitrosti, ki so sfiltrirane zaradi boljse preglednosti
function risia(ime,tdif);
  clc;
  [xfilt,yfilt,vxfilt,vyfilt,vfilt,axfilt,ayfilt,afilt,t]=ata(ime,tdif);
  subplot(221), plot(t,afilt), title('Pospesek')
  subplot(223), plot(t,axfilt), title('Pospesek x')
  subplot(224), plot(t,vyfilt), title('Pospesek y')
end
```

risiv.m

```
% rise Pospeske ki so sfiltrirani zaradi boljse preglednosti

function risia(ime,tdif);
  clc;
  [xfilt,yfilt,vxfilt,vyfilt,vfilt,axfilt,ayfilt,afilt,t]=ata(ime,tdif);
  subplot(221), plot(t,afilt), title('Pospesek')
  subplot(223), plot(t,axfilt), title('Pospesek x')
  subplot(224), plot(t,vyfilt), title('Pospesek y')
end
```

4 Trajektorije gibanja robota

(priloga)