1 Meccanismo per macchina cucitrice

In Fig. 1 è rappresentato lo schema del meccanismo di una macchina cucitrice il cui movente è la manovella O_1A, mentre gli aghi sono montati sui due cedenti O_5H e O_6L. Fissato un sistema di riferimento con origine nel punto O_6, asse x passante per O_1 (positivo da O_6 a O_1) ed asse y verso l'alto, determinare le coordinate del punto L quando la manovella O_1A si trova nella posizione assegnata (si esprima il risultato utilizzando almeno 4 cifre decimali).

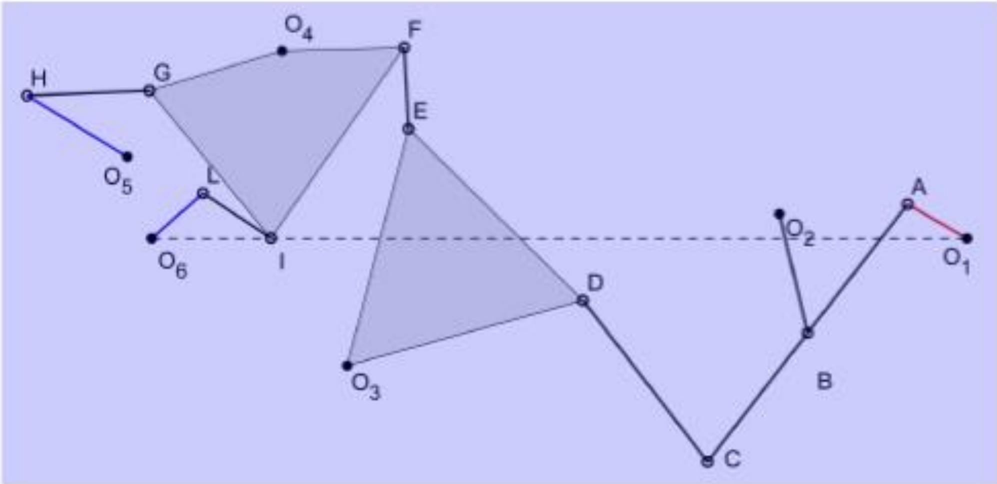


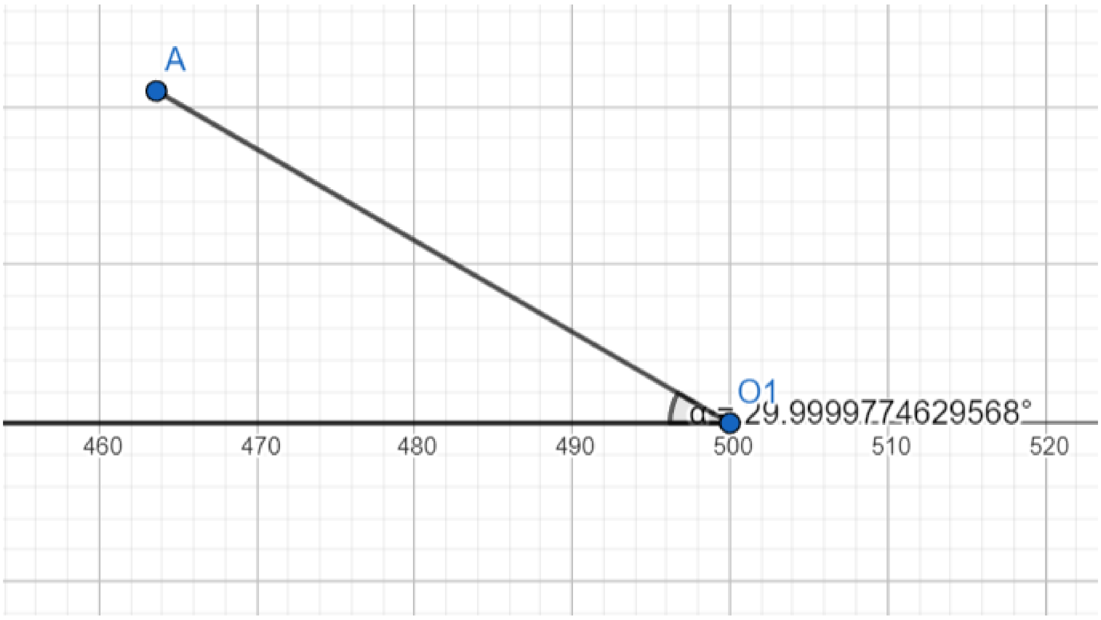
Figura 1: *Meccanismo per macchina cucitrice.*

DATI

u ultima cifra del numero di matricola (matricola = 0000###u)

Posizione delle coppie rotoidali ad asse fisso [mm]	Angoli (misurati in senso antiorario)
O_6 (0, 0) O_1 (500, 0) O_2 (385, 15) O_3 (120, −78) O_4 (80, 115) O_5 (−15, 50)	$D\hat{O}_3E = 60°$ $F\hat{O}_4G = 195°$ $F\hat{O}_4I = 265°$ $A\hat{B}C = 180°$ $A\hat{O}_1O_6 = 30°$
Lunghezze aste [mm] $AB = 100$ $O_2B = 75 - 2 \times u$ $BC = 100$ $CD = 125$ $O_3D = 150$ $O_3E = 150$ $EF = 50$	$O_1A = 42$ $O_4F = 75$ $O4G = 85$ $O_4I = 115$ $GH = 75$ $O_5H = 72$ $IL = 50$ $O_6L = 42$

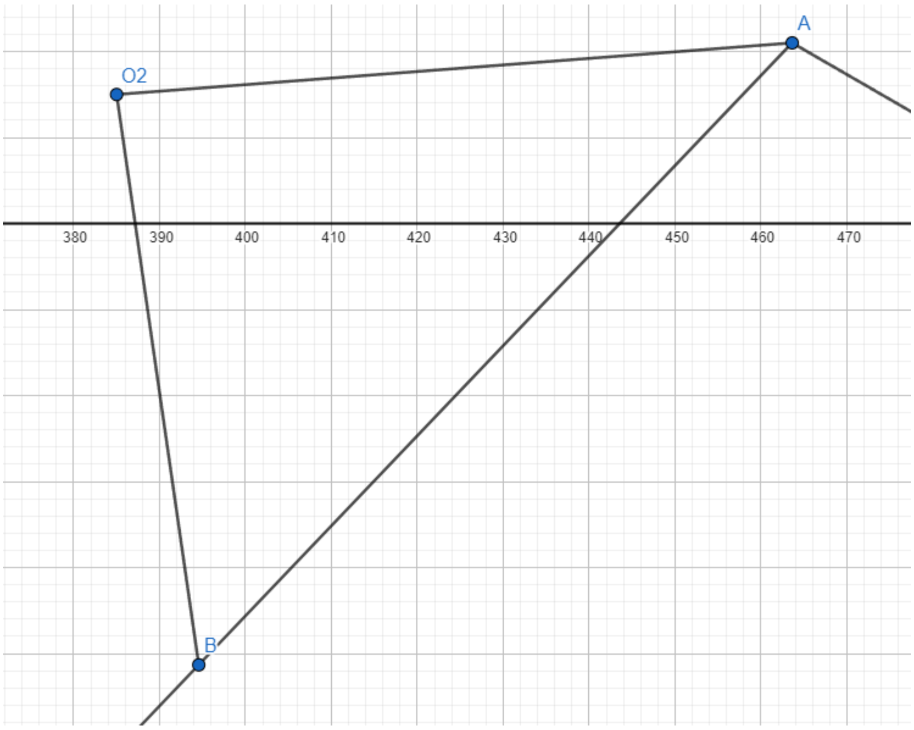
%% GRUPPO DRIVER cerco A



Fissata l'origine in O6 (Per comodità nelle formule verrà chiamato O), noti il punto O1 , l'angolo AO1O6 e la lunghezza dell'asta O1A calcolo il vettore O1A
Trovato il vettore O1A lo sommo al vettore O1 noto e trovo la posizione di A

```
vett_O1A=O1A.*[-cos(ang_AO1O6) , sin(ang_AO1O6)] [mm]
A=O1+vett_O1A [mm]
```

%% DIADE RRR A O2 AB O2B +1



Una diade è un gruppo di Assur composta da tre coppie rotoidali (Un gruppo di Assur è una catena cinematica chiusa avente mobilità nulla, dalla quale non è possibile ottenere un’altra catena cinematica con mobilità nulla se uno o più membri vengono soppressi)

Noti A, O2, |AB|, |AO2| trovo B
Lavoro con configurazione positiva perché per andare da A a O2 trovo B a sinistra

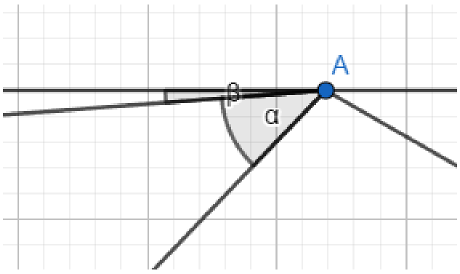
Calcolo O2A usando distanza, una function che mi calcola la distanza fra due punti noti

```
O2A = distanza(O2,A) [mm]
```

Utilizzo le formule della diade RRR per calcolare il vettore (B - A) vettore con coda in A e freccia in B in un sistema di riferimento locale con origine in A

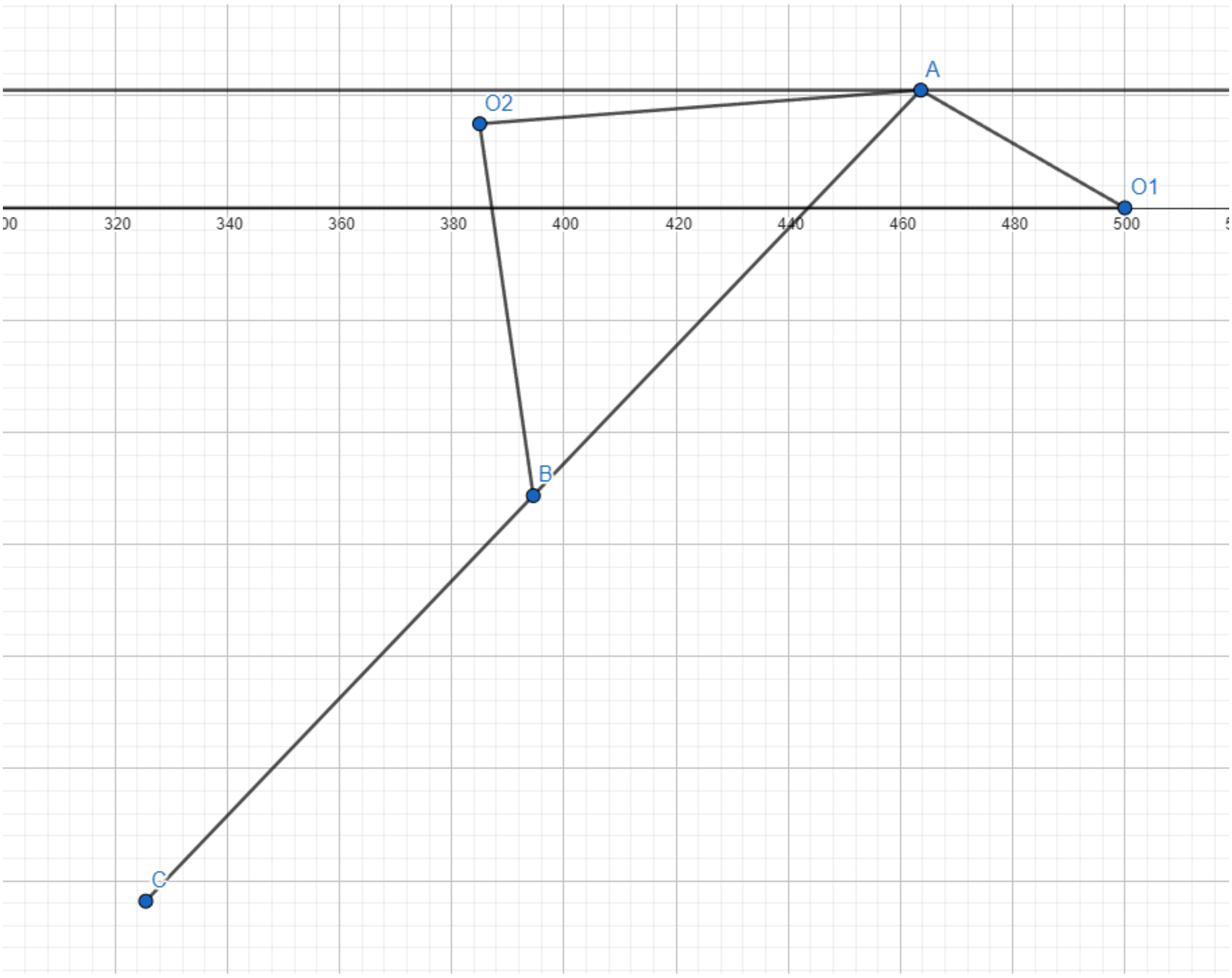
```
lambda1 = 1/2 * (1 + (AB^2 - O2B^2)/(O2A^2) )
mu1 = sqrt ( ( AB^2 / O2A^2 ) - lambda1^2 )
vettloc_AB = [lambda1*O2A , mu1*O2A ]
```

Successivamente per tornare nel sistema di riferimento della consegna calcolo gli angoli alpha e beta rispettivamente 02AB e l’angolo tra il segmento O2A e la parallela all’asse delle ascisse ma con verso opposto all’asse delle x.
Sommando gli angoli alpha e beta trovo l’angolo tra il segmento BA e la parallela all’asse delle ascisse ma con verso opposto all’asse delle x.
Una volta noto gamma calcolo il vettore AB e poi sottraendo al vettore OA il vettore AB trovo il punto B




```
beta = acos(vettloc_AB(x)/AB ) [rad]
alpha = atan( (A(y) - O2(y) )/ O2A ) [rad]
gamma = alpha + beta  rad]
vett_AB= AB * [cos(gamma) , sin(gamma)] [mm]

B=A-vett_AB [mm]
```

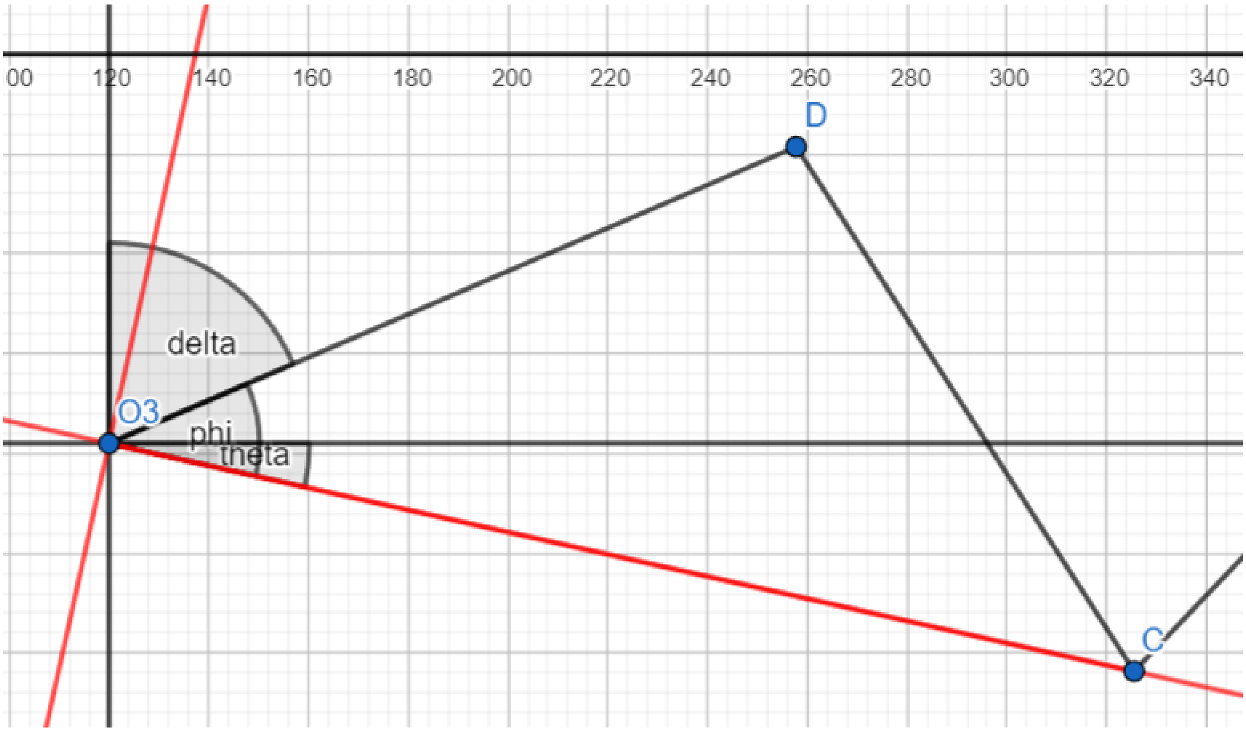
%% ASTA ABC



Trovato B, nota la distanza BC e l’angolo ABC trovo C
Calcolo la distanza AC sommando AB e BC.
Osservando che l’angolo ABC è 180 gradi posso riutilizzare l’angolo gamma per calcolare il vettore AC e di conseguenza trovare il punto C

```
AC= AB+BC [mm]
vett_AC= AC * [cos(gamma) , sin(gamma)] [mm]
C = A - vett_AC [mm]
```

%% DIADE RRR O3 C O3D CD +1



Trovato C e noti O3, distanze O3D e CD cerco D
Scelgo la configurazione positiva perché per andare da O3 a C trovo D alla sinistra
In rosso nella figura ho il sistema di riferimento locale mentre in nero da O3 partono gli assi paralleli agli assi del sistema di riferimento della consegna
Calcolo il vettore O3C e per farlo utilizzo l’angolo theta (che ricavo dalle coordinate di C e O3)

```
teta=  - atan( ( C(y) - O3(y) )/ ( C(x) - O3(x) ) ) [rad]
O3C = ( C(x) - O3(x) ) / cos(teta)   [mm]
```

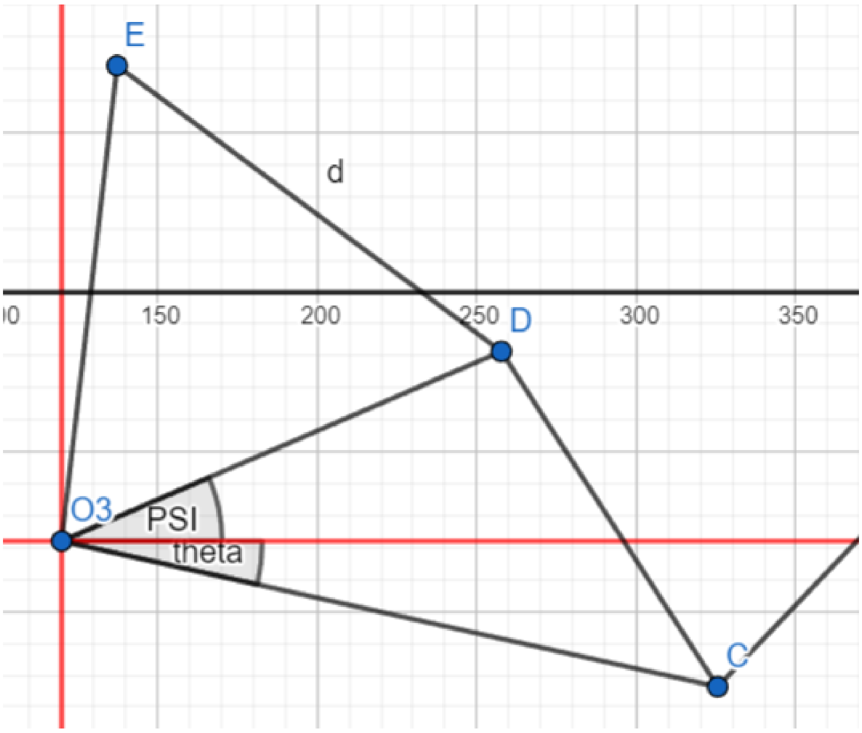
Utilizzando il sistema di riferimento in rosso calcolo il vettore O3D con le formule della diade RRR

```
lambda2 = 1/2 * (1 + (O3D^2 - CD^2)/(O3C^2) )
mu2 = sqrt ( ( O3D^2 / O3C^2 ) - lambda2^2 )
vettloc_O3D = [lambda2*O3C , mu2*O3C ]
```

Ritorno al sistema di riferimento della consegna utilizzando l’angolo pi/2 - delta
Calcolo l’angolo delta sfruttando gli angolo theta e phi(angolo compreso tra il vettore O3D e l’asse x’ del sistema di riferimento in rosso

```
phi=atan( vettloc_O3D(y) / vettloc_O3D(x) ) [rad]
delta= pi/2 - phi + teta [rad]
D = O3 + O3D * [sin(delta) , cos(delta)] [mm]
```

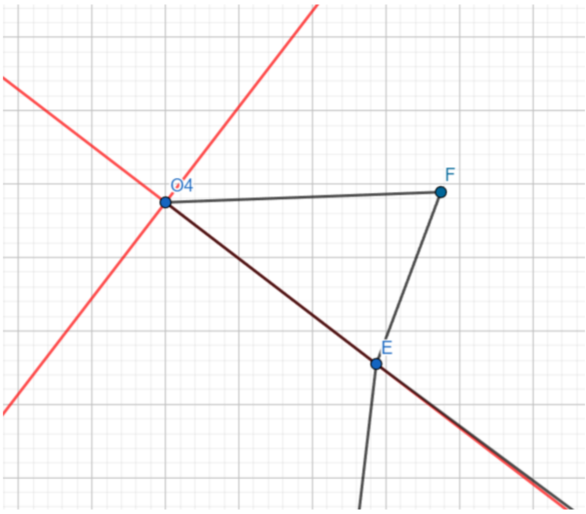
%% TRIANGOLO EQUILATERO O3DE



Noti: D, O3, O3D, O3E e gli angoli theta, phi cerco l'angolo psi compreso tra l'asse parallelo all'asse delle x ma passante per O3 e il segmento O3D
Successivamente notando che il triangolo EO3D è equilatero trovo la posizione di E

```
psi = phi - teta [rad]
vett_O3E=O3E*[cos(pi/3+psi) , sin(pi/3+psi)] [mm]
E=O3+vett_O3E [mm]
```

%% DIADE RRR O4 E O4F EF +1



Trovato E e noti: O4, distanze O4F e EF cerco F

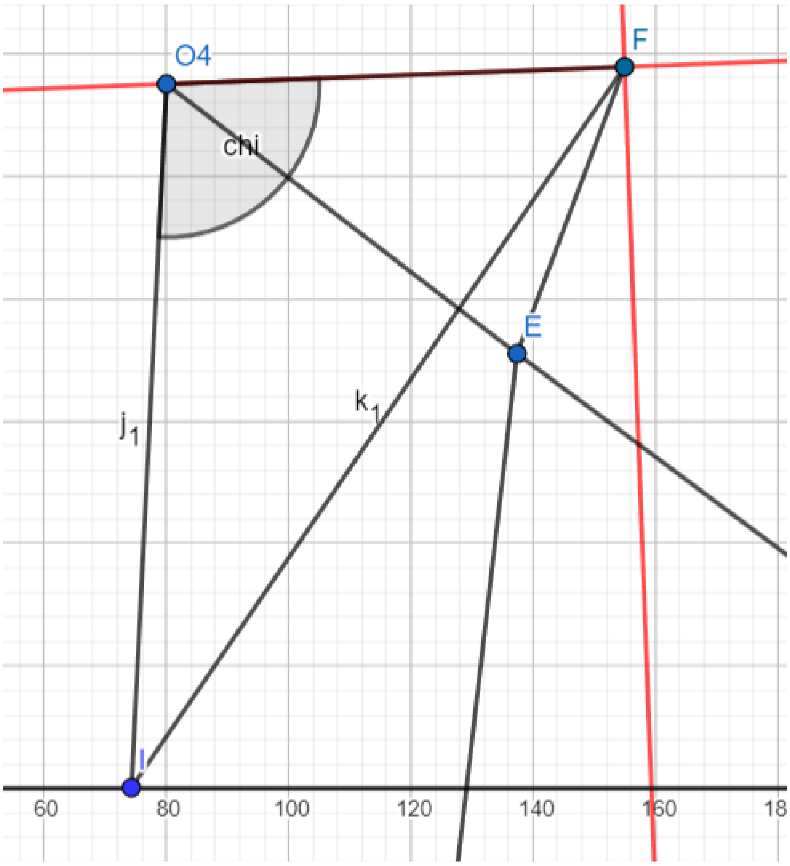
Calcolo la distanza O4 E con la function distanza

```
EO4=distanza(O4,E);
```

Calcolo lambda e mu nel sistema di riferimento in rosso dopodiché passo al sistema di riferimento della consegna per calcolare F

```
lambda3 = 1/2 * (1 + (O4F^2 - EF^2)/(EO4^2) )
mu3 = sqrt ( ( O4F^2 / EO4^2 ) - lambda3^2 )

Fx= lambda3*(E(x) - O4(x)) + mu3* (O4(y) - E(y)) + O4(x) [mm]
Fy= lambda3*(E(y) - O4(y)) + mu3* (E(x) - O4(x)) + O4(y) [mm]
F = [Fx , Fy] [mm]
```

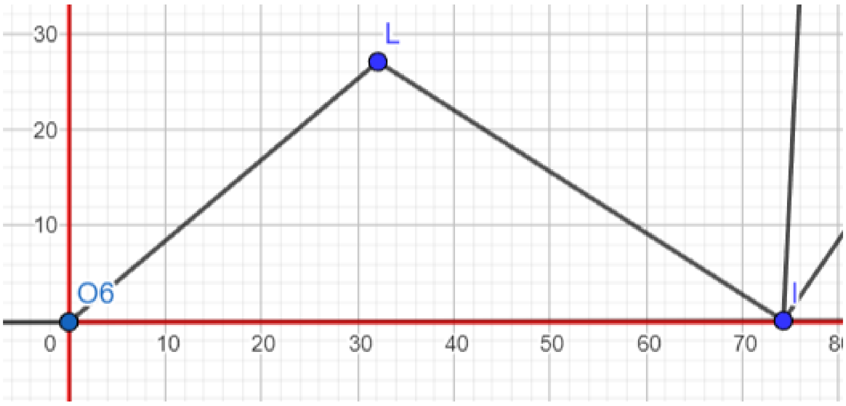
```
%% DIADE RRR F O4 FI O4I +1
```



Calcolo la distanza FI conoscendo: F, l’angolo FO4I e la distanza IO4

L’angolo chi è uguale a 360° - FO4I
chi=2*pi - ang_FO4I [rad]
Vado nel sistema di riferimento in rosso e calcolo: la distanza tra I e F, lambda e mu con le formule della diade RRR

```
IF= sqrt( O4F^2 + O4I^2 -2*O4F*O4I*cos(chi) ) [mm]

lambda4 = 1/2 * (1 + (IF^2 - O4I^2)/(O4F^2) )
mu4 = sqrt ( ( IF^2 / O4F^2 ) - lambda4^2 )

Successivamente sommo i vettori OF e FI e trovo il punto I
Ix= lambda4*( O4(x) - F(x) ) + mu4* ( F(y) - O4(y) ) + F(x)
Iy= lambda4*( O4(y) - F(y) ) + mu4* ( O4(x) - F(x) ) + F(y)
I = [Ix , Iy]

%% DIADE RRR O6 I O6L IL +1
```




```
Analizzo il sistema di riferimento in rosso per calcolare il
vettore O6I con le formule della diade RRR
Utilizzo la function distanza per calcolare la distanza tra
O6 e I
O6I=distanza(O6,I); [mm]

lambda5 = 1/2 * (1 + (O6L^2 - IL^2)/(O6I^2) );
mu5 = sqrt ( ( O6L^2 / O6I^2 ) - lambda5^2 );

Torno, infine, nel sistema di riferimento della consegna e
calcolo Lx e Ly sommando al vettore O6I il vettore O6

Lx= lambda5*(I(x) - O6(x)) + mu5* (O6(y) - I(y)) + O6(x) [mm]
Ly= lambda5*(I(y) - O6(y)) + mu5* (I(x) - O6(x)) + O6(y) [mm]
```

Ultima cifra del numero di matricola		L_x [mm]	L_y [mm]
u= 4		32.1221	27.0586

```
%% DATI

u=4;

%coppie rotoidali [mm]
O6= [ 0   , 0  ];
O1= [ 500 , 0  ];
O2= [ 385 , 15 ];
O3= [ 120 , -78 ];
O4= [ 80  , 115];
O5= [ -15 , 50 ];

%lunghezze aste [mm]
AB=100;
O2B=75-2*u;
BC=100;
CD=125;
O3D=150;
O3E=150;
EF=50;
O1A=42;
O4F=75;
O4G=85;
O4I=115;
GH=75;
O5H=72;
IL=50;
O6L=42;

%angoli [rad]
ang_DO3E=60*pi/180;
ang_FO4G=195*pi/180;
ang_FO4I=265*pi/180;
ang_ABC=180*pi/180;
ang_AO1O6=30*pi/180;

%% GRUPPO DRIVER cerco A

vett_O1A=O1A.*[-cos(ang_AO1O6) , sin(ang_AO1O6)];
A=O1+vett_O1A;

%% DIADE RRR A O2 AB O2B +1

O2A = distanza(O2,A);
%sdr locale
lambda1 = 1/2 * (1 + (AB^2 - O2B^2)/(O2A^2) );
mu1 = sqrt ( ( AB^2 / O2A^2 ) - lambda1^2 );
vettloc_AB = [lambda1*O2A , mu1*O2A ];
```

```
%sdr con origine in A

beta = acos(vettloc_AB(1)/AB ); %[rad]
alpha = atan( (A(2) - O2(2) )/ O2A ); %[rad]
gamma = alpha + beta ;
vett_AB= AB * [cos(gamma) , sin(gamma)];

B=A-vett_AB;

%% ASTA ABC

AC= AB+BC;
vett_AC= AC * [cos(gamma) , sin(gamma)];
C = A - vett_AC;

%% DIADE RRR O3 C O3D CD +1

teta=  - atan( ( C(2) - O3(2) )/ ( C(1) - O3(1) ) ); % [rad]
O3C = ( C(1) - O3(1) ) / cos(teta) ;

%sdr locale

lambda2 = 1/2 * (1 + (O3D^2 - CD^2)/(O3C^2) );
mu2 = sqrt ( ( O3D^2 / O3C^2 ) - lambda2^2 );
vettloc_O3D = [lambda2*O3C , mu2*O3C ];

%sdr originale

phi=atan( vettloc_O3D(2) / vettloc_O3D(1) ); % [rad]
delta= pi/2 - phi + teta; % [rad]
D = O3 + O3D * [sin(delta) , cos(delta)];
%nota bene sin e cos invertiti vedi disegno

%% TRIANGOLO EQUILATERO O3DE

psi = phi - teta;
vett_O3E=O3E*[cos(pi/3+psi) , sin(pi/3+psi)];
E=O3+vett_O3E;

%% DIADE RRR O4 E O4F EF +1

EO4=distanza(O4,E);

lambda3 = 1/2 * (1 + (O4F^2 - EF^2)/(EO4^2) );
mu3 = sqrt ( ( O4F^2 / EO4^2 ) - lambda3^2 );

Fx= lambda3*( E(1) - O4(1) ) + mu3* ( O4(2) - E(2) ) + O4(1);
Fy= lambda3*( E(2) - O4(2) ) + mu3* ( E(1) - O4(1) ) + O4(2);
F = [Fx , Fy];
```

```
%% DIADE RRR F O4 FI O4I +1

chi=2*pi - ang_FO4I;
IF= sqrt( O4F^2 + O4I^2 -2*O4F*O4I*cos(chi) );

lambda4 = 1/2 * (1 + (IF^2 - O4I^2)/(O4F^2) );
mu4 = sqrt ( ( IF^2 / O4F^2 ) - lambda4^2 );

Ix= lambda4*( O4(1) - F(1) ) + mu4* ( F(2) - O4(2) ) + F(1);
Iy= lambda4*( O4(2) - F(2) ) + mu4* ( O4(1) - F(1) ) + F(2);
I = [Ix , Iy];

%% DIADE RRR O6 I O6L IL +1

O6I=distanza(O6,I);

lambda5 = 1/2 * (1 + (O6L^2 - IL^2)/(O6I^2) );
mu5 = sqrt ( ( O6L^2 / O6I^2 ) - lambda5^2 );

Lx= lambda5*( I(1) - O6(1) ) + mu5* ( O6(2) - I(2) ) + O6(1);
Ly= lambda5*( I(2) - O6(2) ) + mu5* ( I(1) - O6(1) ) + O6(2);
L = [Lx , Ly]
```

2 Sintesi cinematica di un quadrilatero articolato

Un punto P di biella di un quadrilatero articolato deve assumere tre posizioni assegnate da occuparsi in tre istanti prefissati. In particolare, a partire dall'istante $t_1 = 0$ in cui il punto di biella è nella posizione P_1, i punti P_2 e P_3 devono essere raggiunti rispettivamente negli istanti t_2 e t_3. Il movente del quadrilatero (la prima asta A_0A della diade di sinistra) ruota con velocità angolare pari a n.

Mediante sintesi cinematica, individuare il quadrilatero. In particolare, con riferimento alla Fig. 2, occorre determinare:

1. le coordinate degli assi delle coppie rotoidali fisse (A_0 e B_0)
2. le lunghezze delle aste accoppiate a telaio (A_0A e B_0B)
3. la distanza tra gli assi delle coppie rotoidali di biella (AB)
4. l'angolo che l'asta A_0A forma con la congiungente gli assi delle coppie fisse quando il punto P è nella prima posizione

Si chiede inoltre di **disegnare il quadrilatero ottenuto in opportuna scala**.

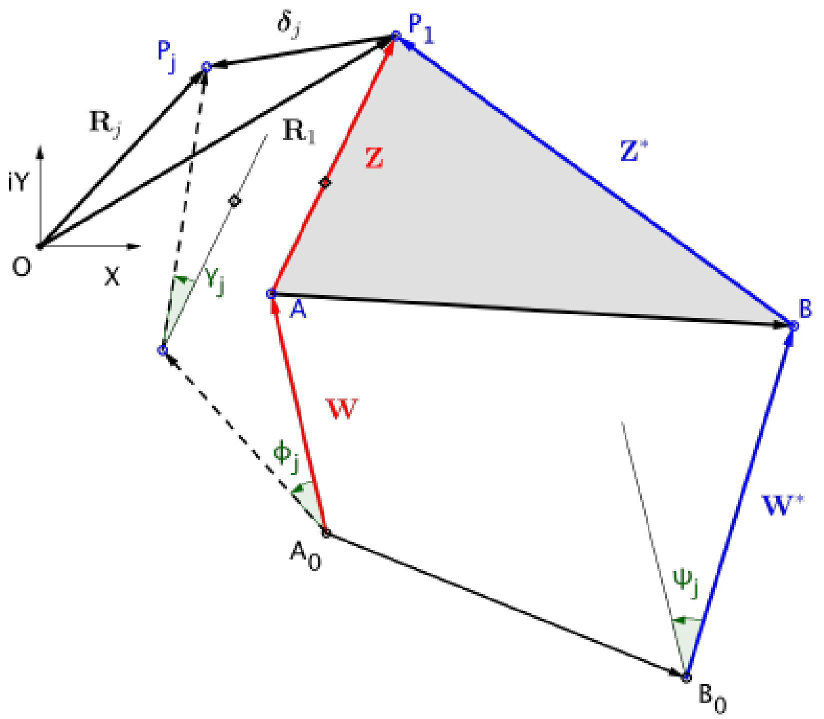


Figura 2: *Schema per la sintesi cinematica del quadrilatero articolato.*

Affinché con i dati forniti il problema abbia una soluzione unica (in modo da consentire il controllo dei risultati da parte del docente), si assuma che quando il punto P occupa rispettivamente le posizioni P_2 e P_3:

1. la prima asta (B_0B) della diade di destra abbia compiuto, a partire dalla posizione iniziale, rotazioni assegnate ψ_2 e ψ_3;
2. un segmento di biella abbia compiuto, a partire dalla posizione iniziale, rotazioni assegnate γ_2 e γ_3.

Oltre alla soluzione così ottenuta, si chiede di riportare anche altre soluzioni (almeno due) scegliendo arbitrariamente gli angoli ψ_j e γ_j. I quadrilateri corrispondenti a tali ulteriori soluzioni vanno disegnati in scala opportuna e comparati con quello ottenuto dalla prima soluzione.

DATI

u è il resto della divisione per 4 del numero di matricola.

P_1	$(200, -75)$	mm
P_2	$(25 \times u + 60, 25 \times u - 151)$	mm
P_3	$(25 \times u + 100, 25 \times u - 305)$	mm
t_2	1.05	s
t_3	2.1	s
n	20	rpm
ψ_2	33	deg
ψ_3	37	deg
γ_2	-10	deg
γ_3	45	deg

```
%% ANGOLI DI INCLINAZIONE DELL'ASTA AoA
```

Sono noti i punti P2, P3 e gli istanti in cui l'asta si deve portare nelle posizioni indicate t2 e t3. Cerco gli angoli di inclinazione dell'asta AoA (phi2 phi3) nel portarsi nelle posizioni successive.

```
%L'asta AoA si porta nelle posizione A2 e A3

delta2= P2- P1; %[mm]
delta3= P3- P1; %[mm]

velocità angolare asta AoA
w_asta_AoA= n*2*pi/60; %[rad/s]

A-->A1
phi1= w_asta_AoA*t1; %[rad]
```

```
A-->A2
phi2= w_asta_AoA*t2; %[rad]
A-->A3
phi3= w_asta_AoA*t3; %[rad]


%% DIADE DI SINISTRA

Analizzo la catena di sinistra in rosso e cerco W e Z
Utilizzo la formula della diade standard due volte una per
ogni configurazione

1) W*(e^i*phi2 - 1) + Z*( e^i*gamma2 - 1) = delta2
2) W*(e^i*phi3 - 1) + Z*( e^i*gamma3 - 1) = delta3

Ho 2 equazioni vettoriali in 4 incognite: Wx,Wy,Zx,Zy . Sono
noti phi2, phi3, delta2, delta3 , gamma2 e gamma3.

Risolvo applicando la regola di Cramer
W= det(Matrice a numeratore W)/det(Matrice a denominatore W);
```

Matrice a Numeratore W:

delta2	e^i*gamma2 - 1
delta3	e^i*gamma3 - 1

Matrice a denominatore W:

e^i*phi2 - 1	e^i*gamma2 - 1
e^i*phi3 - 1	e^i*gamma3 - 1

```
Z= det(Matrice a numeratore Z)/det(Matrice a denominatore Z);
```

Matrice a Numeratore Z:

e^i*phi2 - 1	delta2
e^i*phi3 - 1	delta3

Matrice a denominatore Z:

e^i*phi2 - 1	e^i*gamma2 - 1
e^i*phi3 - 1	e^i*gamma3 - 1

```
numW = [ delta2 , (cos(gamma2) +1i*sin(gamma2) - 1) ; delta3
, (cos(gamma3) +1i*sin(gamma3) - 1) ];
denW = [(cos(phi2) +1i*sin(phi2) - 1) , (cos(gamma2)
+1i*sin(gamma2) - 1) ; (cos(phi3) +1i*sin(phi3) - 1),
(cos(gamma3) +1i*sin(gamma3) - 1) ];

W= det(numW) / det(denW); [mm]

numZ= [(cos(phi2) +1i*sin(phi2) - 1) , delta2 ; (cos(phi3)
+1i*sin(phi3) - 1), delta3 ];
```

```
denZ= [(cos(phi2) +1i*sin(phi2) - 1) , (cos(gamma2)
+1i*sin(gamma2) - 1); (cos(phi3) +1i*sin(phi3) - 1),
(cos(gamma3) +1i*sin(gamma3) - 1) ];

Z= det(numZ) / det(denZ); [mm]

%% CATENA DI DESTRA
```

Stesso procedimento per la catena di destra in blu ma cerco W* e Z*

1) W_asterisco*(e^i*psi2 - 1) + Z_asterisco*(e^i*gamma2 - 1) = delta2
2) W_asterisco*(e^i*psi3 - 1) + Z_asterisco*(e^i*gamma3 - 1) = delta3

Risolvo applicando la regola di Cramer
W= det(Matrice a numeratore W*)/
/det(Matrice a denominatore W*);

Matrice a Numeratore W*:

delta2	e^i*gamma2 - 1
delta3	e^i*gamma3 - 1

Matrice a denominatore W*:

e^i*psi2 - 1	e^i*gamma2 - 1
e^i*psi3 - 1	e^i*gamma3 - 1

Z= det(Matrice a numeratore Z)/det(Matrice a denominatore Z);

Matrice a Numeratore Z:

e^i*psi2 - 1	delta2
e^i*psi3 - 1	delta3

Matrice a denominatore Z:

e^i*psi2 - 1	e^i*gamma2 - 1
e^i*psi3 - 1	e^i*gamma3 - 1

```
numW_asterisco = [ delta2 , (cos(gamma2) +1i*sin(gamma2) - 1)
; delta3 , (cos(gamma3) +1i*sin(gamma3) - 1) ];
denW_asterisco = [(cos(psi2) +1i*sin(psi2) - 1) ,
(cos(gamma2) +1i*sin(gamma2) - 1) ; (cos(psi3) +1i*sin(psi3)
- 1), (cos(gamma3) +1i*sin(gamma3) - 1) ];

W_asterisco= det(numW_asterisco) / det(denW_asterisco); [mm]

numZ_asterisco= [(cos(psi2) +1i*sin(psi2) - 1) , delta2 ;
(cos(psi3) +1i*sin(psi3) - 1), delta3 ];
```

```
denZ_asterisco= [(cos(psi2) +1i*sin(psi2) - 1) , (cos(gamma2)
+1i*sin(gamma2) - 1); (cos(psi3) +1i*sin(psi3) - 1),
(cos(gamma3) +1i*sin(gamma3) - 1) ];

Z_asterisco= det(numZ_asterisco) / det(denZ_asterisco); [mm]


%% COPPIE ROTOIDALI FISSE A0 B0
```

Calcolo le coordinate degli assi delle coppie rotoidali fisse noti : P1,Z, W, Z* e W*

A0 = P1-Z-W [mm]
B0 = P1-Z*-W* [mm]

```
A0 = P1-Z-W
B0 = P1-Z*-W*

%% LUNGHEZZA ASTE ACCOPPIATE A TELAIO
```

Calcolo la lunghezza delle aste AoA e BoB noti W e W*

AoA = $\sqrt{Wx^2+Wy^2}$ [mm]

BoB = $\sqrt{W^*x^2+W^*y^2}$ [mm]

```
A0A=sqrt( real(W)^2 + imag(W)^2 );
B0B=sqrt( real(W_asterisco)^2 + imag(W_asterisco)^2 );

%% DISTANZA TRA LE COPPIE ROTOIDALI DI BIELLA
```

Calcolo la distanza tra gli assi delle coppie rotoidali di biella

OA = P1-Z ; [mm]
OB = P1-Z*; [mm]

AB = $\sqrt{(\mathrm{OA}x-OBx)^2+(\mathrm{OA}y-OBy)^2}$ [mm]

```
OA = P1-Z ;
OB = P1-Z_asterisco;
AB = sqrt((real(OA)-real(OB))^2+(imag(OA)-imag(OB))^2);
A=OA
B=OB
```

%% angolo tra asta A0A E A0B0

Calcolo l'angolo che l'asta AoA forma con la congiungente agli assi delle coppie fisse quando P è nella prima posizione

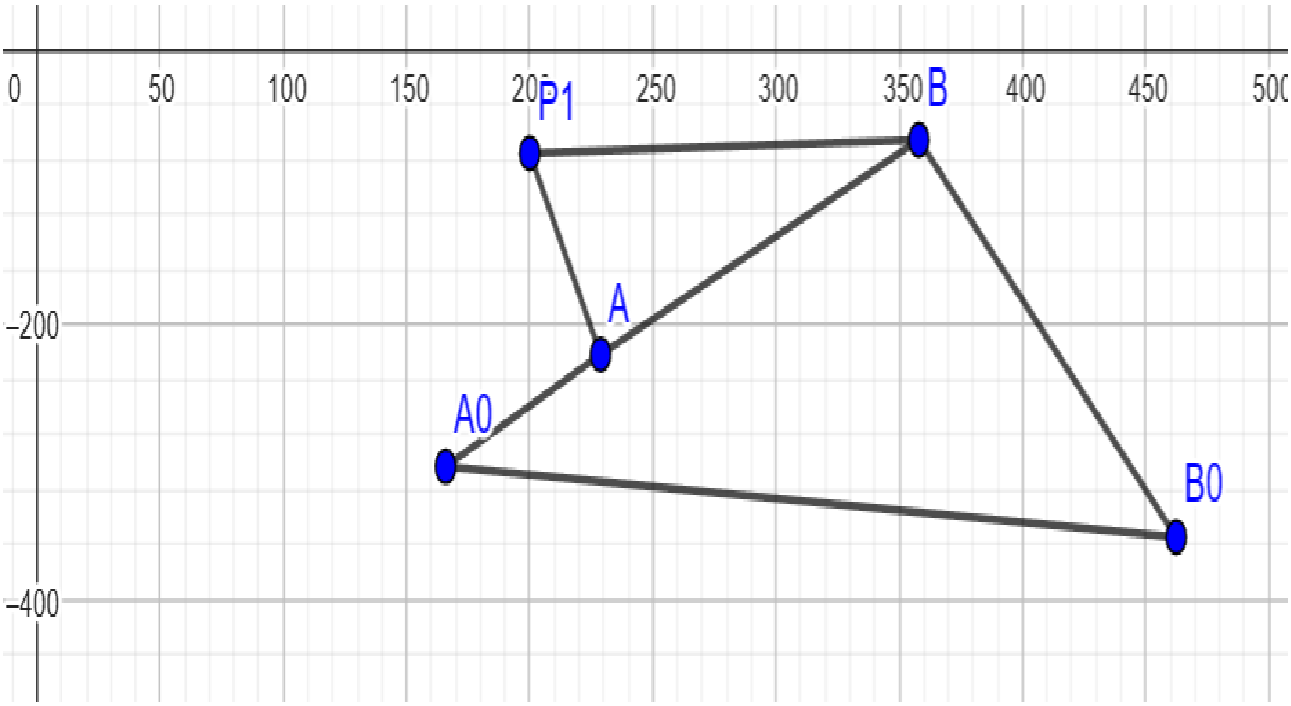
Alfa = arctan((Ao - Bo)y / (Ao - Bo)x) [°]

Beta = arctan((A-Ao)y / (A-Ao)x) [°]

Gamma = alfa + beta [°]

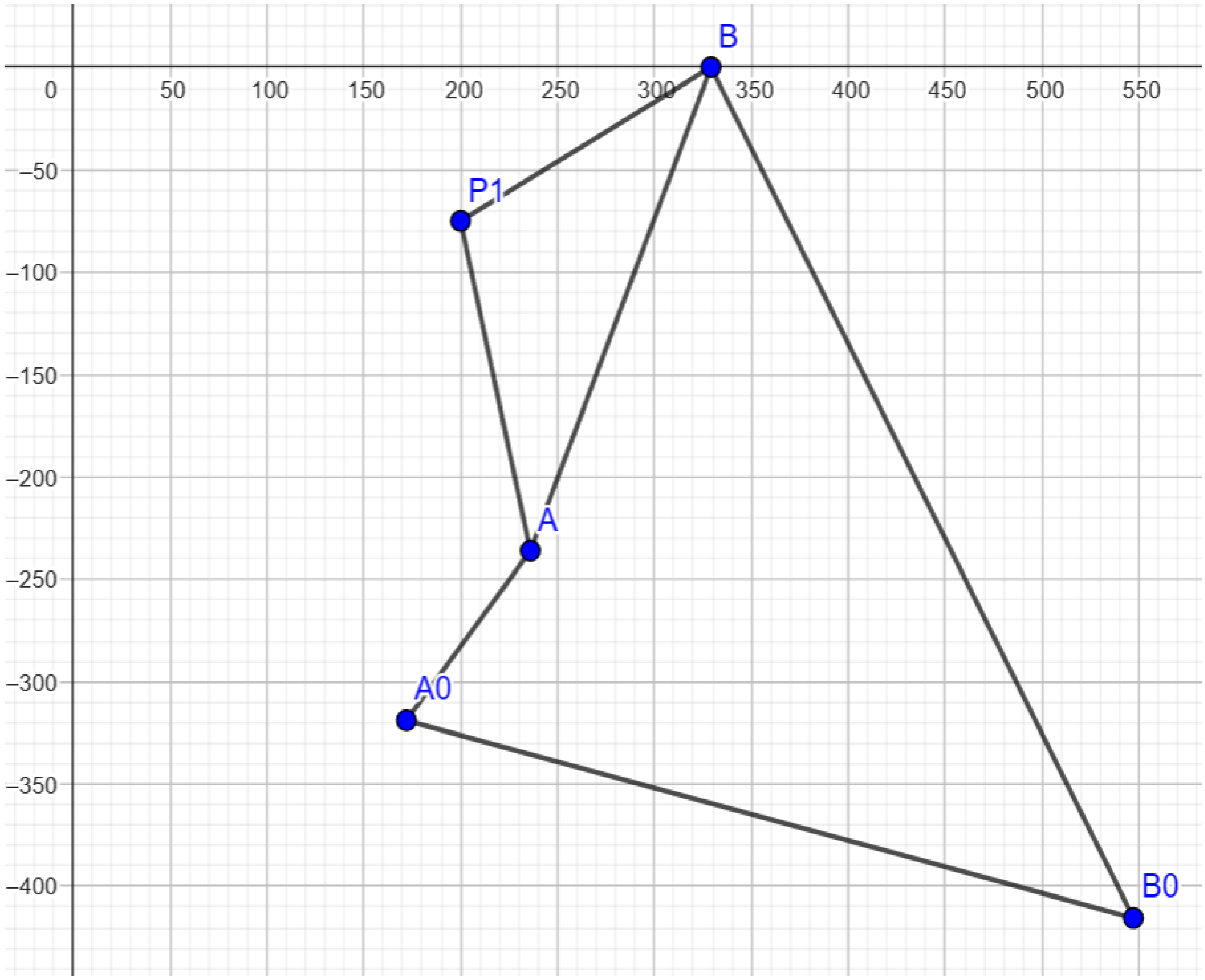
```
alfa = atand((imag(A0)-imag(B0))/(real(A0)-real(B0)));
beta = atand((imag(OA)-imag(A0))/(real(OA)-real(A0)));
gamma = abs(alfa)+abs(beta);
```

Disegno il quadrilatero utilizzando geogebra



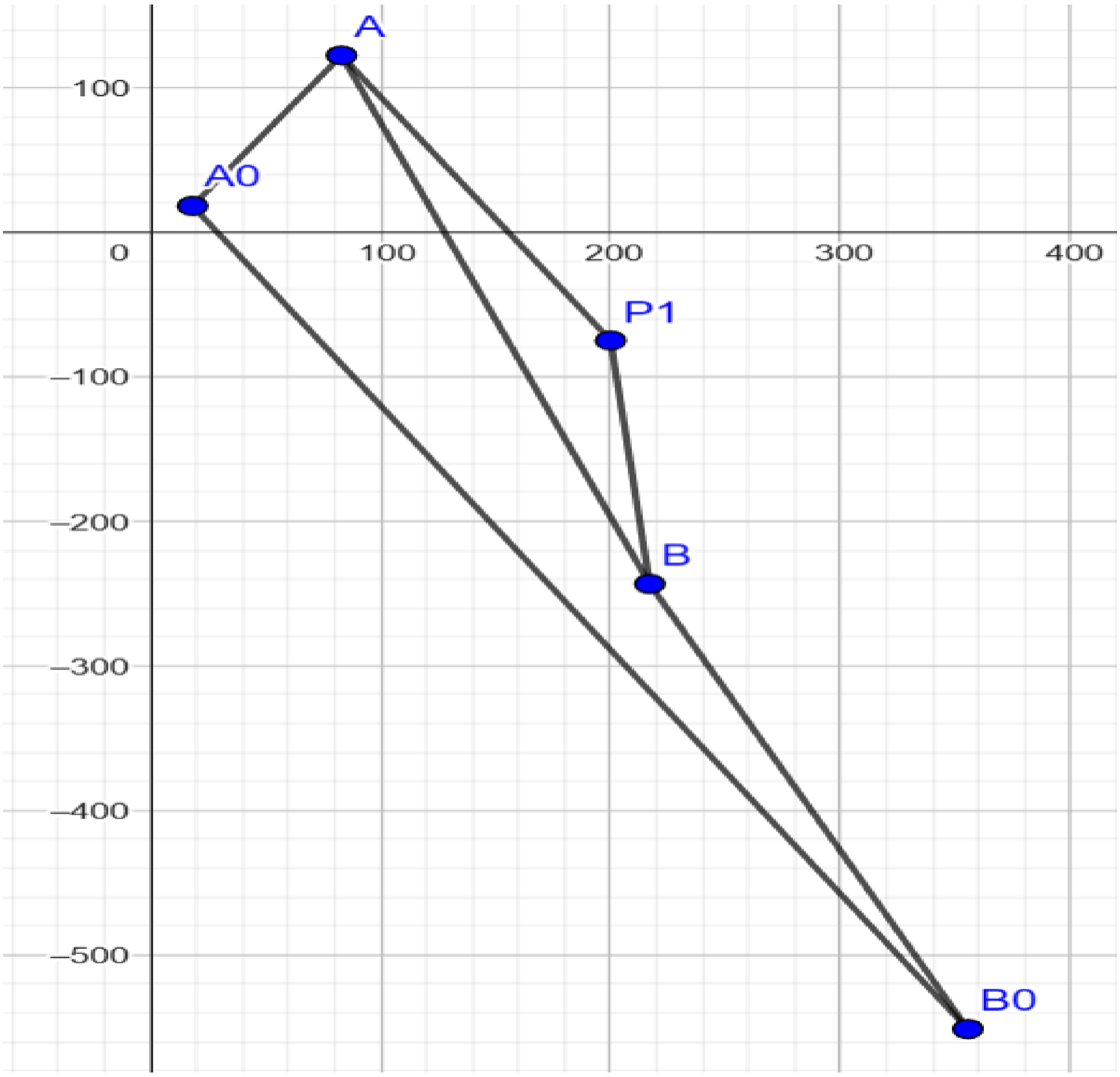
L'unità di misura degli assi del sistema di riferimento utilizzato su geogebra è il millimetro

```
Modifico arbitrariamente i dati e rilancio il programma:
psi2 = 20*pi/180; [rad]
psi3 = 30*pi/180;  [rad]
gamma2 = -10*pi/180;  [rad]
gamma3 = 40*pi/180;  [rad]
```



L'unità di misura degli assi del sistema di riferimento utilizzato su geogebra è il millimetro

```
Modifico arbitrariamente i dati per una seconda volta e
rilancio il programma:
psi2 = 20*pi/180; [rad]
psi3 = 50*pi/180;  [rad]
gamma2 = 15*pi/180;  [rad]
gamma3 = -30*pi/180;  [rad]
```



SINTESI CINEMATICA DI UN QUADRILATERO ARTICOLATO

Resto[1]	Coordinate punto A_0 $[mm]$	Coordinate punto B_0 $[mm]$	Lunghezza asta A_0A $[mm]$	Lunghezza biella AB $[mm]$	Lunghezza asta B_0B $[mm]$	Angolo iniziale manovella $[deg]$
0	165,8594 - 303,3744i	462,4280 - 354,6821 i	103,0640	202,8438	307,6107	62.2100

```
u=3; % è il resto della divisione per 4 del numero di
matricola

%% DATI

P1 = 200-75i; % [mm]
P2 = (25 * u + 60)+(25*u - 151)*1i; % [mm]
P3 = (25 * u + 100)+ (25 * u - 305)*1i; % [mm]
t1=0; %[s]
t2 = 1.05; %[s]
t3 = 2.1; %[s]
n = 20; %[rpm]
psi2 = 33*pi/180; %[rad]
psi3 = 37*pi/180; % [rad]
gamma2 = -10*pi/180; % [rad]
gamma3 = 45*pi/180; % [rad]

%% ANGOLI DI INCLINAZIONE DELL'ASTA AoA

%L'asta AoA si porta nelle posizione A2 e A3

delta2= P2- P1; %[mm]
delta3= P3- P1; %[mm]

w_asta_AoA= n*2*pi/60; %[rad/s]
phi1= w_asta_AoA*t1; %[rad]
phi2= w_asta_AoA*t2; %[rad] A-->A2
phi3= w_asta_AoA*t3; %[rad] A-->A3

%% DIADE DI SINISTRA

% Ho 2 equazioni vettoriali in 6 incognite. Devo perciò
scegliere
% arbitrariamente 2 valori e avere così 4 equazioni in 4
incognite
% scelgo phi2 e phi3 --> ho come incognite W e Z

% 1) W*(e^i*phi2 - 1) + Z*( e^i*gamma2 - 1) = delta2
% 2) W*(e^i*phi3 - 1) + Z*( e^i*gamma3 - 1) = delta3

% ricorda e^i*x= cosx+isinx)
% CRAMER :

numW = [ delta2 , (cos(gamma2) +1i*sin(gamma2) - 1) ; delta3
, (cos(gamma3) +1i*sin(gamma3) - 1) ];
denW = [(cos(phi2) +1i*sin(phi2) - 1) , (cos(gamma2)
+1i*sin(gamma2) - 1) ; (cos(phi3) +1i*sin(phi3) - 1),
(cos(gamma3) +1i*sin(gamma3) - 1) ];
```

```
W= det(numW) / det(denW); %[mm]

numZ= [(cos(phi2) +1i*sin(phi2) - 1) , delta2 ; (cos(phi3)
+1i*sin(phi3) - 1), delta3 ];
denZ= [(cos(phi2) +1i*sin(phi2) - 1) , (cos(gamma2)
+1i*sin(gamma2) - 1); (cos(phi3) +1i*sin(phi3) - 1),
(cos(gamma3) +1i*sin(gamma3) - 1) ];

Z= det(numZ) / det(denZ); %[mm]

%% CATENA DI DESTRA

% 1) W_asterisco*(e^i*psi2 - 1) + Z_asterisco*( e^i*gamma2 -
1) = delta2
% 2) W_asterisco*(e^i*psi3 - 1) + Z_asterisco*( e^i*gamma3 -
1) = delta3

% CRAMER :

numW_asterisco = [ delta2 , (cos(gamma2) +1i*sin(gamma2) - 1)
; delta3 , (cos(gamma3) +1i*sin(gamma3) - 1) ];
denW_asterisco = [(cos(psi2) +1i*sin(psi2) - 1) ,
(cos(gamma2) +1i*sin(gamma2) - 1) ; (cos(psi3) +1i*sin(psi3)
- 1), (cos(gamma3) +1i*sin(gamma3) - 1) ];

W_asterisco= det(numW_asterisco) / det(denW_asterisco); %[mm]

numZ_asterisco= [(cos(psi2) +1i*sin(psi2) - 1) , delta2 ;
(cos(psi3) +1i*sin(psi3) - 1), delta3 ];
denZ_asterisco= [(cos(psi2) +1i*sin(psi2) - 1) , (cos(gamma2)
+1i*sin(gamma2) - 1); (cos(psi3) +1i*sin(psi3) - 1),
(cos(gamma3) +1i*sin(gamma3) - 1) ];

Z_asterisco= det(numZ_asterisco) / det(denZ_asterisco); %[mm]

%% COPPIE ROTOIDALI FISSE A0 B0

A0 = P1-Z-W;
B0 = P1-Z_asterisco-W_asterisco;

%% LUNGHEZZA ASTE ACCOPPIATE A TELAIO

A0A=sqrt( real(W)^2 + imag(W)^2 );
B0B=sqrt( real(W_asterisco)^2 + imag(W_asterisco)^2 );

%% DISTANZA TRA LE COPPIE ROTOIDALI DI BIELLA
OA = P1-Z ;
OB = P1-Z_asterisco;
```

```
AB = sqrt((real(OA)-real(OB))^2+(imag(OA)-imag(OB))^2);

%% angolo tra asta A0A E A0B0
alfa = atand((imag(A0)-imag(B0))/(real(A0)-real(B0)));
beta = atand((imag(OA)-imag(A0))/(real(OA)-real(A0)));
gamma = abs(alfa)+abs(beta);
```

3 Proporzionamento di un meccanismo a camma

Si consideri il cinematismo di azionamento delle valvole di aspirazione di un motore per autoveicolo rappresentato in Fig. 3: la camma (1), ruotando con velocità angolare pari alla metà della velocità angolare dell'albero motore, comanda il bicchierino (2), che può supporsi solidale con la valvola (3).

Durante la fase di azionamento della valvola il contatto tra camma e bicchierino è assicurato dall'azione della molla (4). La stessa molla garantisce, a valvola chiusa, la tenuta sulla sede (5).

Mentre a valvola aperta la posizione istantanea dell'equipaggio traslante è controllata dal contatto con la camma (1), a valvola chiusa la posizione dello stesso equipaggio è definita dal contatto con la sede (5). Nel passaggio dalle configurazioni di apertura alla configurazione di chiusura (e viceversa) il contatto con l'equipaggio traslante si deve pertanto spostare dalla camma (1) alla sede (5).

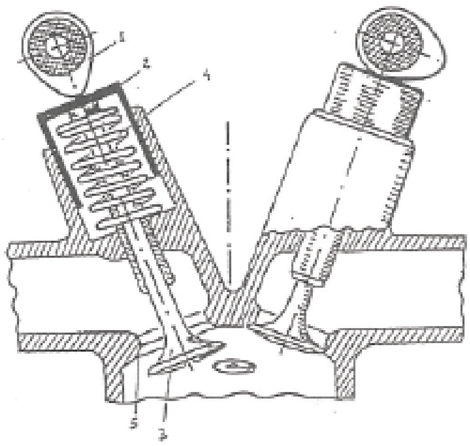


Figura 3: *Cinematismo di azionamento delle valvole di aspirazione di un motore per autoveicolo.*

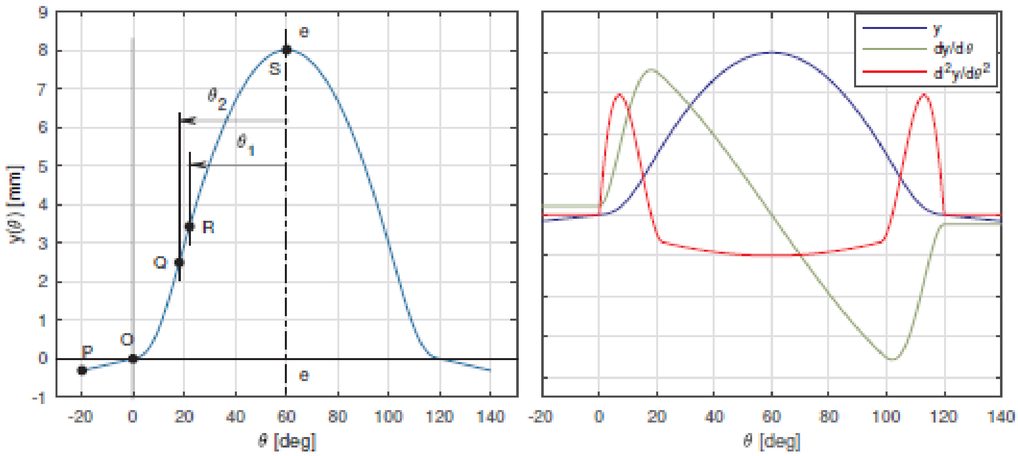
Per assicurare un funzionamento corretto è quindi necessario garantire, a valvola chiusa, un seppur minimo gioco tra camma e bicchierino. Tale gioco può essere impostato a motore freddo (gioco nominale g), ma non può essere controllato e mantenuto rigorosamente costante durante il funzionamento. Per tale motivo il profilo attivo della camma inizia e termina con rampe di recupero gioco aventi la funzione di controllare l'entità degli inevitabili urti tra camma e bicchierino (inizio della fase di apertura valvola) e tra valvola e sede (termine della fase di chiusura valvola) in corrispondenza di un determinato campo di possibili valori del gioco di funzionamento.

Detto θ l'angolo di rotazione della camma, si richiede di:

1. determinare la legge del moto $y = y(\theta)$ della valvola tenendo presente i seguenti vincoli progettuali:
   (a) legge di moto simmetrica rispetto alla configurazione di massima apertura;
   (b) alzata della valvola pari ad h (al netto delle rampe di ripresa del gioco);
   (c) durata di azionamento valvola corrispondente ad α gradi di rotazione dell'albero motore (al netto delle rampe di ripresa del gioco);

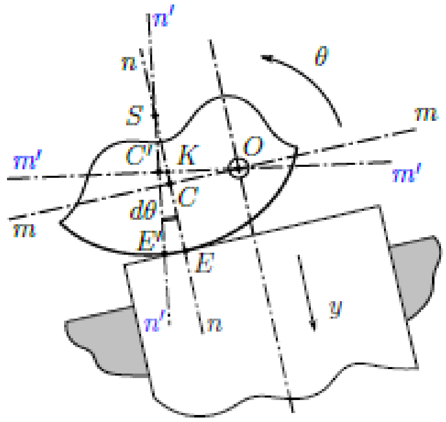
   (d) altezza rampe di ripresa gioco $= h_1$;
   (e) massima velocità d'urto $= u$;
   (f) massimo regime di rotazione del motore $= n$;
   (g) rampe di recupero gioco a velocità di impatto costante al variare del gioco di funzionamento;
   (h) tratto utile della legge di moto composto da tre archi polinomiali di ordine 5, 4 e 5, raccordati tra loro a distanza θ_1 dalla configurazione di massima apertura;
   (i) continuità di $y'''(\theta)$ (ossia di $d^3y/d\theta^3$) in corrispondenza dei suddetti raccordi;
   (j) continuità di $y''(\theta)$ in corrispondenza dei raccordi tra gli archi polinomiali di ordine 5 e le rampe di recupero gioco;
   (k) accelerazione valvola nulla a distanza θ_2 dalla configurazione di massima apertura.

2. tracciare gli andamenti di y, y' e y'' in funzione di θ;
3. determinare il raggio base della camma in modo tale che il minimo raggio di curvatura del profilo sia pari a δ_{min} (arrotondare al decimo di mm);
4. disegnare la forma della camma (se ne esegua il disegno in scala 2:1);
5. determinare la massima distanza dall'asse valvola del contatto camma-bicchierino (l'asse dell'albero a camme e del bicchierino sono ortogonali ed incidenti);
6. determinare il diametro minimo d del bicchierino, essendo pari a $d/2$ lo spessore della camma;
7. determinare l'entità del precarico della molla a valvola chiusa e in condizioni di gioco nominale $g = h_1$ affinché, in corrispondenza del massimo regime di rotazione del motore, la forza minima di contatto tra camma e bicchierino sia pari a F_{min}; è nota la rigidezza k della molla e la massa m dell'equipaggio mobile della valvola.

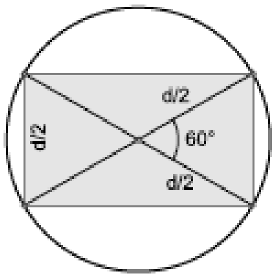


(a) (b)

Figura 4: *a) Legge di alzata; b) Curve $y(\theta)$, $y'(\theta)$, $y''(\theta)$.*



(a)



(b)

Figura 5: *a) Schema cinematico; b) Dimensionamento del bicchierino.*

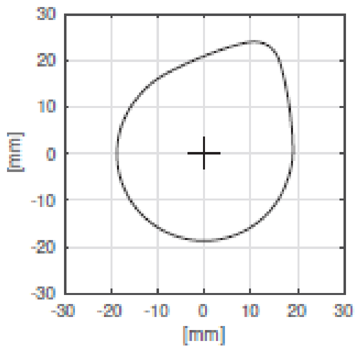


Figura 6: *Profilo della camma.*

DATI

Resto della divisione per 4 del numero di matricola	**0**	**1**	**2**	**3**
h [mm] alzata al netto delle rampe di ripresa gioco	8	8	10	10
α [°]	240	240	252	252
h_1 [mm] altezza rampe di ripresa gioco	0.3	0.3	0.3	0.3
u [mm/s] massima velocità d'urto	270	270	270	270
n [rpm] massimo regime di rotazione del motore	6000	6000	5400	5400
θ_1 [°]	38	38	37	37
θ_2 [°]	42	42	42	42
δ_{min} [mm] minimo raggio di curvatura del profilo	5	6	5	6
F_{min} [N] forza minima di contatto	140	120	150	130
k [N/mm] rigidezza della molla	8.8	8.3	10	9.5
m [kg] massa equipaggio mobile della valvola	0.19	0.21	0.22	0.20

Meccanismo a Camma punteria piattello in cui voglio del gioco tra la camma e il bicchierino solidale. Devo inizialmente determinare la legge di moto tenendo conto dei vincoli progettuali.

Otterrò così un sistema sotto forma di matrice contenente le 9 condizioni

```
%% calcoli
```

In questo primo paragrafo del mio esercizio imposto le condizioni per analizzare la legge di moto

Cerco la massima velocità di urto y°max ricordando che dal momento che il tratto di rampa ha pendenza costante (y'=y'o)
→ y'= y°max/ omega_max)
Calcolo omegamax e noto y°max = u ricavo y'

```
omega_camma= ( n/2 )* (2*pi/60 ) ; [rad/s]
omegamax = 2pi*n/(60*2) = pi*n/60 [rpm] -->
omegamax = pi*n/60 * 180/pi=3n [deg/s]
omegamax = 3*n; %[deg/s]

y'= u/omegamax [mm/s / deg/s] --> [mm/deg]
```

Cerco l'ascissa dei punti P,Q,R,S :

```
tq= alpha/4-theta2;       [deg]
thetaR= alpha/4-theta1;   [deg]
thetaS= alpha/4;          [deg]
thetaP = -h1/y';          [deg]
```

Analizzo la zona attiva. Essa è divisa in 3 tratti :

1) P – O lineare

2) O – R polinomio di 5 grado:

```
Ya(theta) =a5*theta^5 + a4*theta^4 + a3*theta^3 +
+a2*theta^2 + a1*theta + a0

ya'= 5*a5*theta^4 + 4*a4*theta^3 + 3*a3*theta^2 +
2*a2*theta + a1

ya''= 20*a5*theta^3 + 12*a4*theta^2 + 6*a3*theta +
2*a2

ya'''= 60*a5*theta^2 + 24*a4*theta + 6*a3
```

3) R – S polinomio di 4 grado:

Yb= b4*(thetaRS-thetaS)^4 +b2*(thetaRS-thetaS)^2+ b0

yb'= 4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS)

yb''= 12*b4*(thetaRS-thetaS)^2 + 2*b2

yb'''= 24*b4*(thetaRS-thetaS)

Dalle mie condizioni sulla legge di moto posso affermare che:
ya(theta=0) = 0 → a0 = 0
ya’ (theta=0) = y’0 → a1 = y’0
ya’’ (theta=0) = 0 → a2 = 0
ya’’ (thetaQ) = 0
ya(thetaR) = yb(thetaR)
ya’ (thetaR) = yb’(thetaR)
ya’’ (thetaR) = yb’’(thetaR)
ya’’’ (thetaR) = yb’’’(thetaR)
yb(thetaS) = h → bo=h

Rimangono 5 equazioni in 5 incognite a3,a4,a5,b2,b4
Ricavo le equazioni in forma analitica e poi metto il problema sotto forma di matrici.

ya’’(thetaQ) = 0

ya’’=20*a5*thetaQ^3+12a4*thetaQ^2+6a3*thetaQ+2a2
- ya’’=20+12a4*thetaQ^2+6a3*thetaQ

ya(thetaR) = yb(thetaR)

a5*thetaR^5 + a4*thetaR^4 + a3*thetaR^3 + a2*thetaR^2 + a1*thetaR + a0 = b4*(thetaR-thetaS)^4 +b3*(thetaR-thetaS)^3 + b2*(thetaR-thetaS)^2+ +b1*(thetaR-thetaS)+ b0

- a5*thetaR^5 + a4*thetaR^4 + a3*thetaR^3 -b4*(thetaR-thetaS)^4 - b2*(thetaR-thetaS)^2 = -yo’*thetaR+bo ya’ (thetaR) = yb’(thetaR)

5*a5*thetaR^4 + 4*a4*thetaR^3 + 3*a3*thetaR^2 + 2*a2*thetaR + a1 = 4*b4*(thetaR-thetaS)^3 + 2*b2*(thetaR-thetaS)

- 5*a5*thetaR^4 + 4*a4*thetaR^3 + 3*a3*thetaR^2 - 4*b4*(thetaR-thetaS)^3 - 2*b2*(thetaR-thetaS) = -yo’

ya’’ (thetaR) = yb’’(thetaR)

- 20*a5*theta^3 + 12*a4*theta^2 + 6*a3*theta – -12*b4*(thetaR-thetaS)^2 -2*b2 =0

ya’’’ (thetaR) = yb’’’(thetaR)

- 60*a5*theta^2 + 24*a4*theta + 6*a3 -24*b4*(thetaR-thetaS) = 0

A= matrice dei coefficienti , {X} = incognite, {b} = vettore dei termini noti. [A]*{X}= {b}

[A]=

20*thetaQ^3	12*thetaQ^2	6*thetaQ	0	0
thetaR^5	thetaR^4	thetaR^3	-(thetaR -thetaS)^4	-(thetaR -thetaS)^2
5* thetaR^4	4*thetaR^3	3*thetaR^2	-4*(thetaR -thetaS)^3	-2*(thetaR -thetaS)
20*thetaR^3	12*thetaR^2	6*thetaR	-12*(thetaR -thetaS)^2	-2
60*thetaR^2	24*thetaR	6	-24*(thetaR -thetaS)	0

{X} =

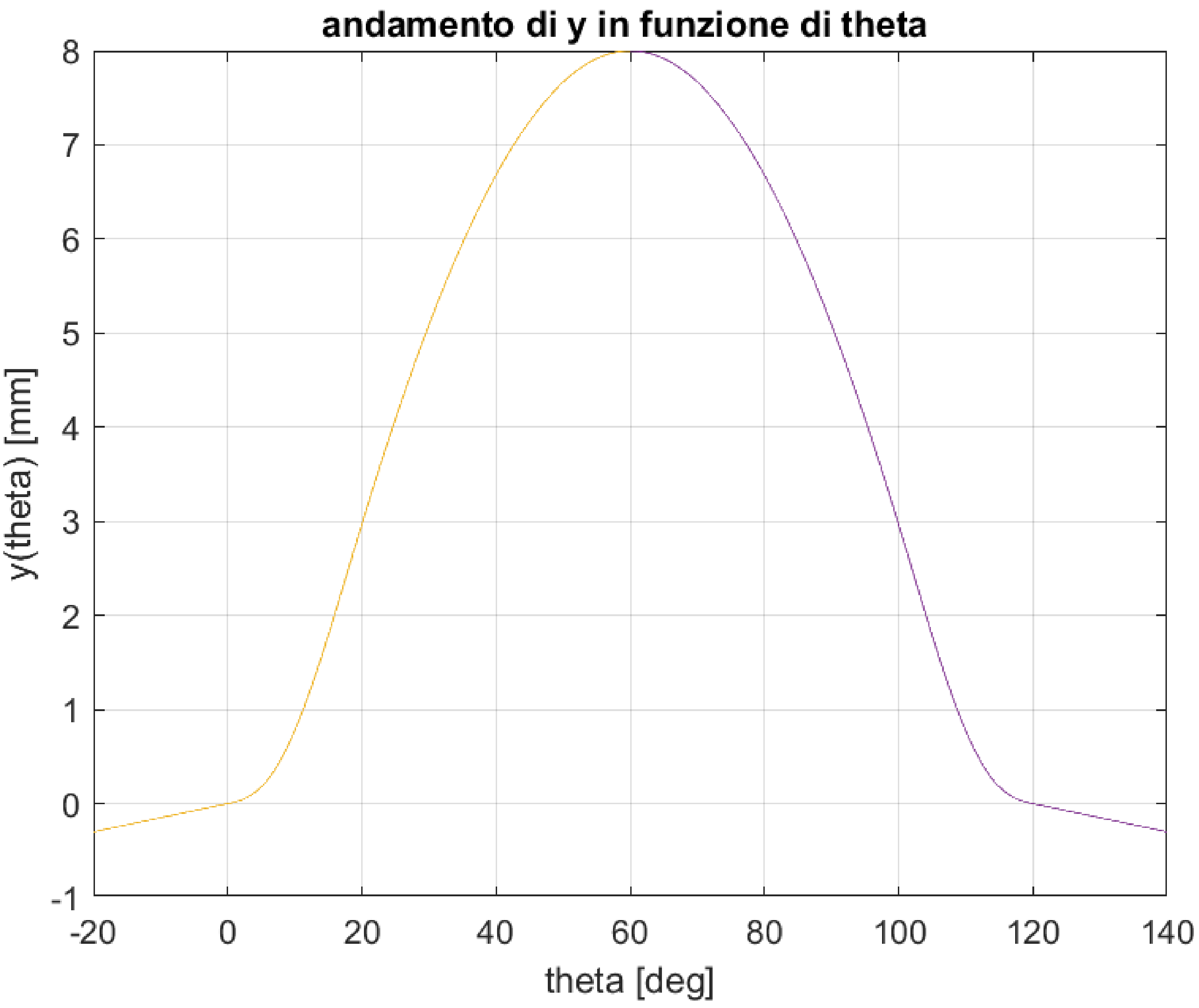
a5
a4
a3
a2
a1

{b} =

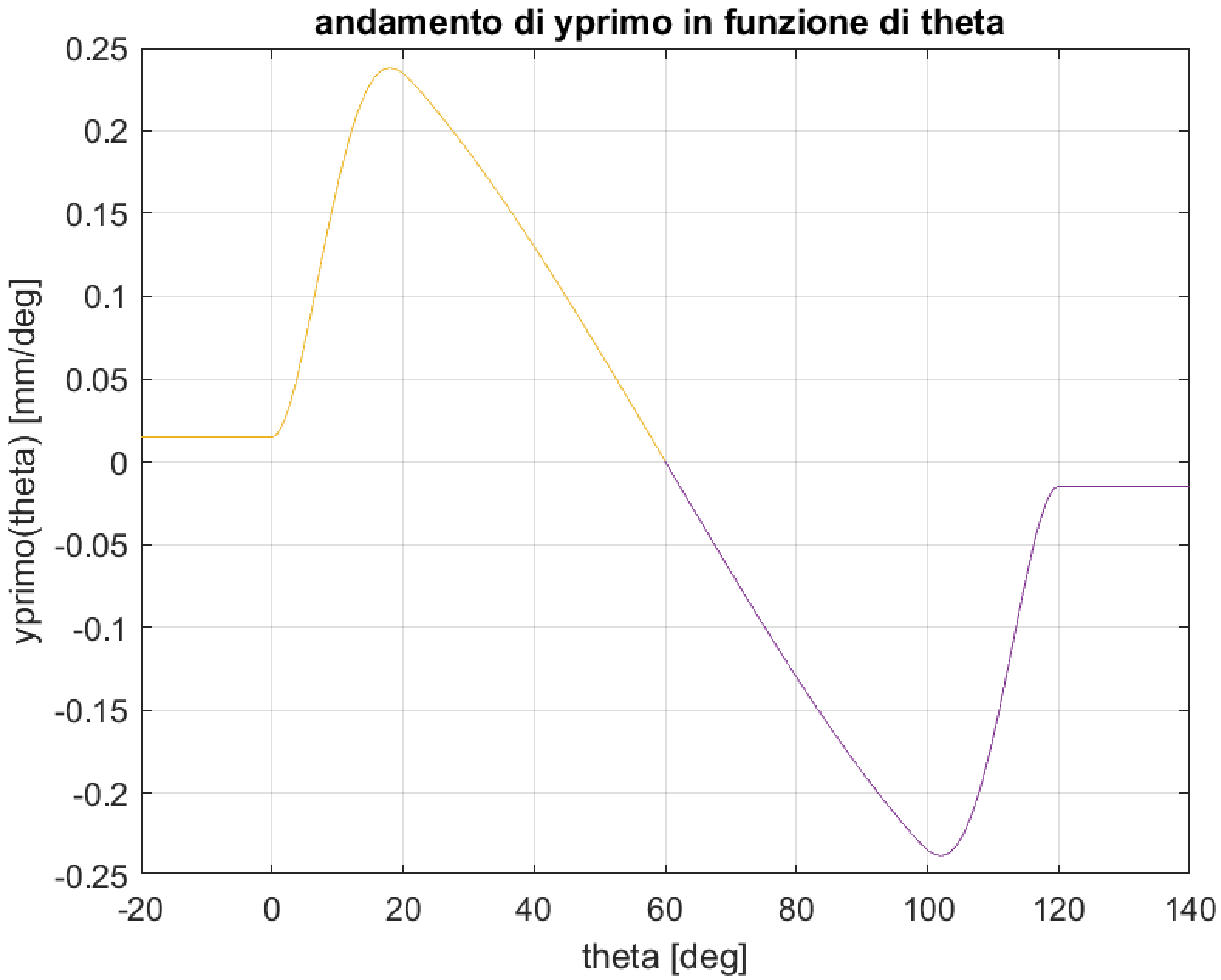
0
bo – y’o *thetaR
– y’o
0
0

```
format shortE

A=[20*tq^3 12*tq^2 6*tq  0 0; thetaR^5 thetaR^4 thetaR^3 -(thetaR-thetaS)^4 -(thetaR-thetaS)^2; 5*thetaR^4  4*thetaR^3 3*thetaR^2 -4*(thetaR-thetaS)^3 -2*(thetaR-thetaS); 20*thetaR^3 12*thetaR^2 6*thetaR -12*(thetaR-thetaS)^2 -2; 60*thetaR^2 24*thetaR 6 -24*(thetaR-thetaS) 0];
```

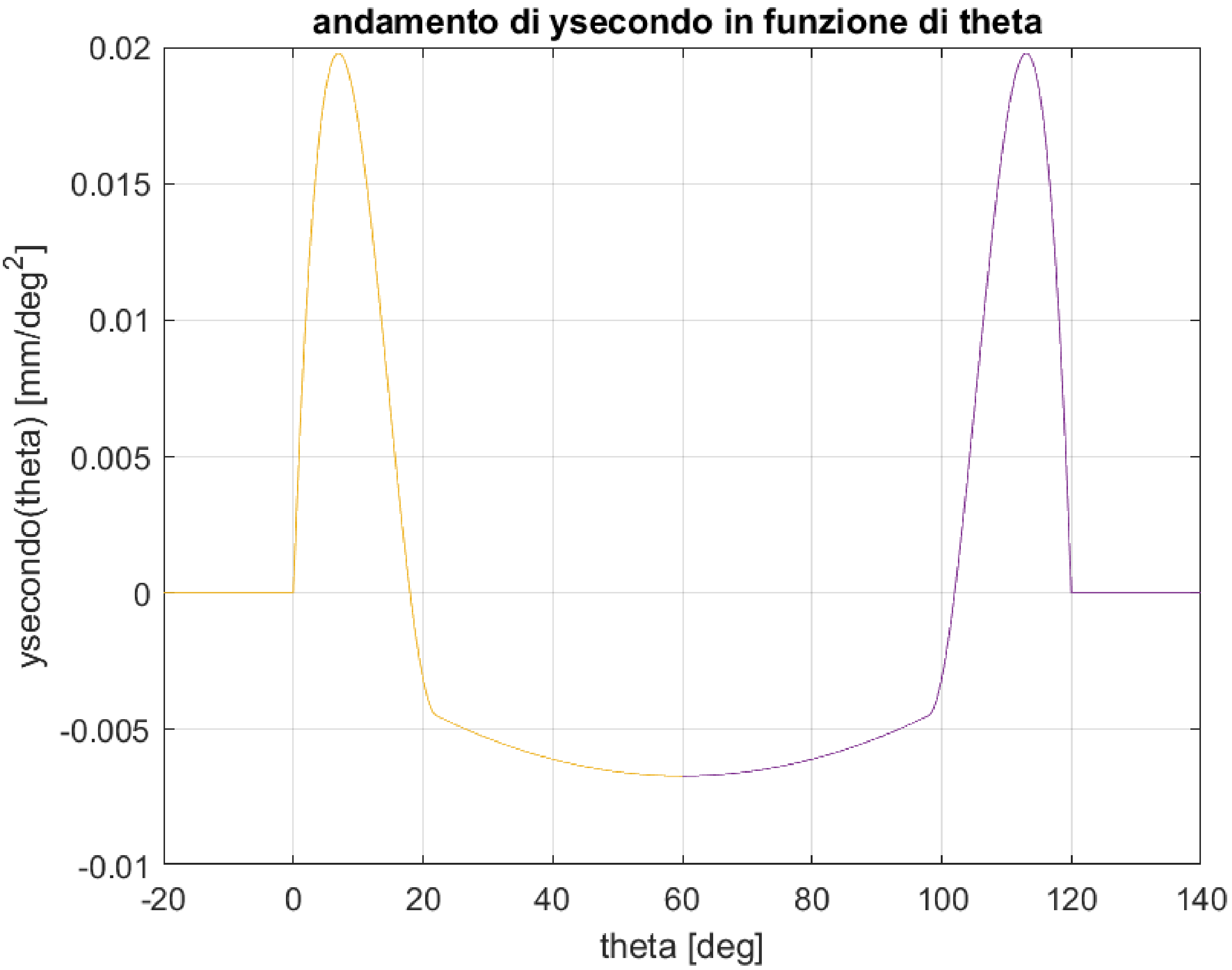
```
b=[0; b0-(yprimo*thetaR); -yprimo; 0; 0];

x=A\b;

a5=x(1);
a4=x(2);
a3=x(3);
b4=x(4);
b2=x(5);
a0=0;
a1=yprimo;
a2=0;

%% andamento di y in funzione di theta
```



```
%% andamento di y' in funzione di theta
```



```
%% andamento di y' in funzione di theta
```




```
%% determinare il raggio base della camma

il minimo raggio di curvatura è pari deltamin

rho = Rb + h1 + y + y'' >= deltamin
z(theta)= y + y''
cerco il minimo di rho --> d( z(theta) ) / dtheta = 0

ho rischio maggiore di avere un rho basso nella zona che è
più appuntita perchè la circonferenza osculatrice è più
piccola. La zona più appuntita è quella rappresentata dal
polinomio di quarto grado R - S
chiamo yb yRS per non sovrascrivere le variabili
```

Ricordandomi che:

    yb= b4*(thetaRS-thetaS)^4 + b2*(thetaRS-thetaS)^2+ b0
    yb'= 4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS)
    yb''= 12*b4*(thetaRS-thetaS)^2 + 2*b2
    yb'''= 24*b4*(thetaRS-thetaS)

```
d(yb + yb'')/dtheta = yb'+yb'''=
= 4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS) +
+24*b4*(thetaRS-thetaS) =
= (thetaRS-thetaS)*[ 4*b4*(thetaRS-thetaS)^2 + 2*b2 + 24*b4 ]
```

Imponendo la derivata nulla ho due soluzioni:

```
1) thetaRS = thetaS

2) 4*b4*(thetaRS-thetaS)^2 + 2*b2 + 24*b4 = 0

4*b4*thetaRS^2 + 4*b4*thetaS^2 -8*b4*thetaRS*thetaS + 2*b2 +
24*b4 = 0

[ 4*b4*thetaRS^2 ] + [ - 8*b4*thetaS*thetaRS ] +
+ [ b4(4*thetaS^2 + 24)+ 2b2 ]= 0

thetaRS1,2 = { 8*b4*thetaS +- sqrt{64*(b4^2)*(thetaS^2) -
4*[(b4*(4*thetaS^2 + 24) + 2*b2 )]*4b4} } / (8*b4)

delta = 64*(b4^2)*(thetaS^2) - 4*[(b4*(4*thetaS^2 + 24) +
2*b2 )]*4b4 = 64*(b4^2)*(thetaS^2) - 64*(b4^2)*(thetaS^2) -
384*b4^2 - 32*b2*b4 = - 384*b4^2 - 32*b2*b4 < 0
```

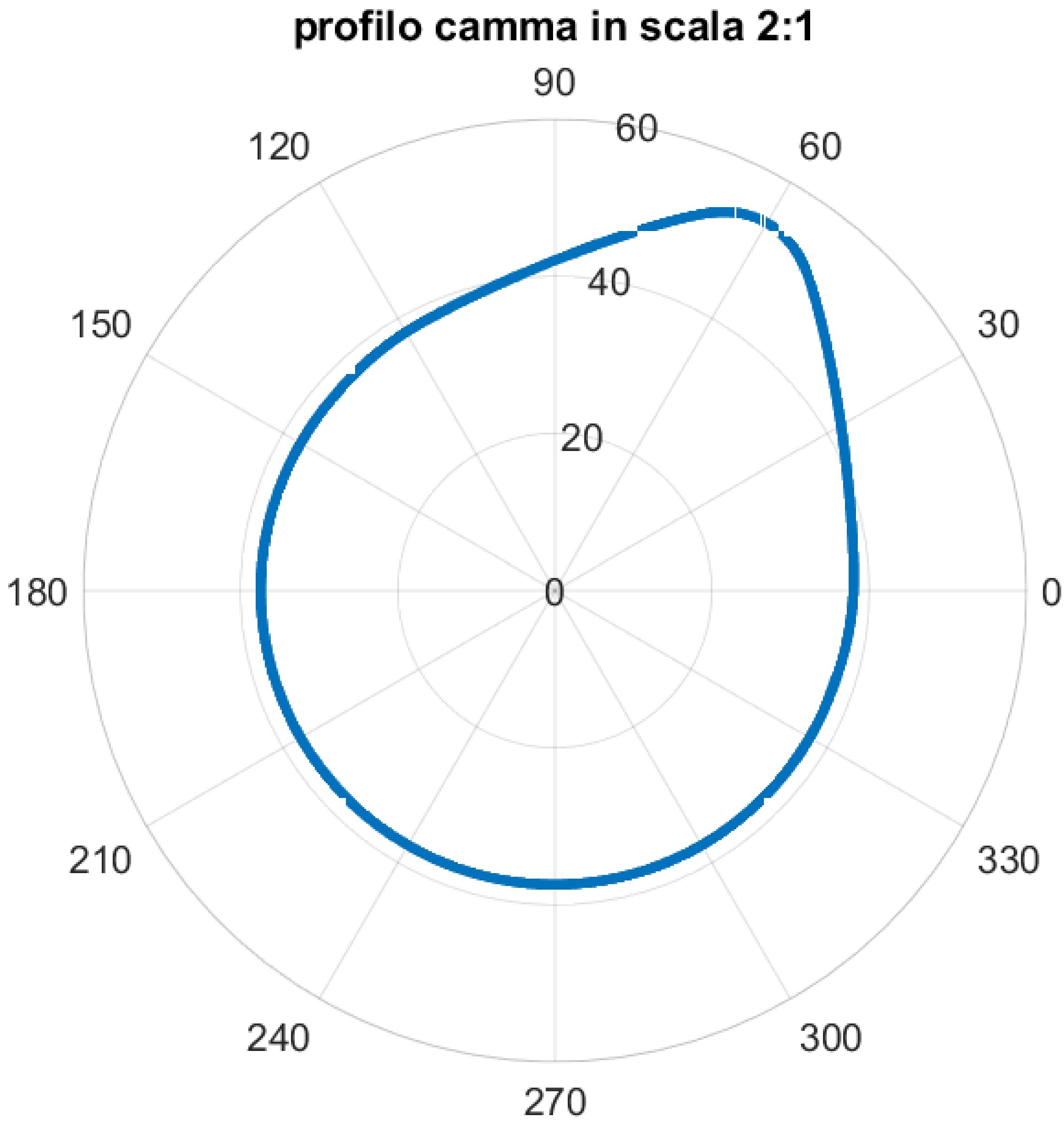
il delta è negativo quindi avrei valori non reali perciò
escludo la seconda soluzione

```
rhomin = Rb + h1 + z(thetaRS = thetaS) = deltamin
```

Metto a sistema:

```
1) Rb =  deltamin - h1 - z(thetaRS = thetaS)
2) z = y + y''
3) yb=b4*(thetaRS-thetaS)^4 + b2*(thetaRS-thetaS)^2+ b0
4) yb''=  12*b4*(thetaRS-thetaS)^2 + 2*b2


Rb =  deltamin - h1 - yb(thetaS) - yb''(thetaS)  =
Rb = deltamin - h1 - b0 - 2*b2
[mm] - [mm] - [mm] - [mm/deg^2] → devo moltiplicare b2 per
*(360/2pi)^2 per avere la stessa unità di misura

Rb = deltamin - h1 - b0 - 2*b2*( (360/(2*pi))^2 ); [mm]
```

%% disegnare il profilo della camma in scala 2:1



%% bicchierino

determino la massima distanza dall'asse valvola del contatto camma-bicchierino

devo avere sempre contatto fra camma e bicchierino quindi il bicchierino deve avere un ampiezza almeno uguale a y'max

OCmax=max([OC1 OC2a OC2b OC3]); [mm]
Max è una funzione che mi trova il valore massimo di un vettore (notazione: y'max= OCmax)
ricorda : OC = y'=dy/dtheta quindi devo correggere l'unità di misura se voglio una distanza in mm

```
determino il diametro minimo d del bicchierino essendo pari a
d/2 lo spessore della camma

d/2= y'max / cos(30) [mm]

d= 2 * OCmax / cos(pi/6); [mm]

%% precarico molla

F = Fel + Fin  + T; [N]
T = Precarico [N]
Fel = Felastica [N]
Fel = - ky  [N]
Fin = Finerziale [N]
Fin = - m*d^2y/dt^2 = -m*coefacc*omega^2 [N]

Cerco la Fmin Fmin =  (Fel + Fin)min  + T; [N]

il minimo si raggiunge sul polinomio yb (zona R - S )
(Fel + Fin)min = - (ky + m*coefacc*omega^2 )
[N/mm] * [mm] + [kg * mm / s^2 * 1/1000]

devo trovare il minimo di - (k*yRS + m*coefaccRS*omegamax^2 )
d( (Fel + Fin) ) / dtheta = 0
d ( (k*yRS + m*coefaccRS*omegamax^2 ) ) = 0
k*yb' + m* yb''' * omegamax^2 = 0

ricavate precedentemente (zona R - S) :
     yb= b4*(thetaRS-thetaS)^4 + b2*(thetaRS-thetaS)^2+ b0
     yb'= 4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS)
     yb''=  12*b4*(thetaRS-thetaS)^2 + 2*b2
     yb'''= 24*b4*(thetaRS-thetaS)

k*(4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS) ) +
m*omegamax^2*24*b4*(thetaRS-thetaS)= 0

(thetaRS-thetaS)*[4k*b4*(thetaRS-thetaS)^2 +2k*b2
+24m*omegamax^2*b4] = 0
ho due soluzioni
(thetaRS è theta ho dovuto aggiungere un pedice per non
confondermi)
1) theta = thetaS

2) 4k*b4*(theta-thetaS)^2 +2k*b2 + 24m*omegamax^2*b4 = 0
```

```
4k*b4*theta^2 + 4k*b4*thetaS^2 - 8k*b4*theta*thetaS + 2k*b2
+24m*omegamax^2*b4 = 0

4k*b4*(theta^2) - 8k*b4*thetaS*(theta) + (4k*b4*thetaS^2
+2k*b2 + 24m*omegamax^2*b4 ) = 0

theta1,2=  8k*b4*thetaS +- sqrt (64*k^2*b4^2*thetaS^2 -
4*(4k*b4*thetaS^2+2k*b2+24m*omegamax^2*b4 )*(4k*b4))

delta = 64*k^2*b4^2*thetaS^2 -
4*(4k*b4*thetaS^2+2k*b2+24m*omegamax^2*b4)*(4k*b4)
=64*k^2*b4^2*thetaS^2 - 4^3*b4^2*k^2*thetaS^2 -
4^2*2*k^2*b2*b4 - 4^2*24*m*omegamax^2*b4^2 =  - 32*k^2*b2*b4
- 384*m*omegamax^2*b4^2 < 0

delta < 0 --> le soluzioni non sono reali
utilizzo perciò la 1 soluzione

1) theta = thetaS
T = Fmin - k*yb(thetaS) - m*coefaccRS(thetaS)*omegamax^2

T = Fmin - k*bo - m*2b2*omegamax^2

Fmin = [N]
k = [N /mm]
bo = [mm]
omegamax=3*n;  [mm/deg^2]
b2 = [deg^2/s^2]
m= massa equipaggio mobile della valvola [kg]

[N] - [N /mm] * [mm] - [kg]*[deg^2/s^2]*[mm/deg^2]
[N] - [N ] - [kg*mm/s^2]
[N] - [N ] - [N*1/1000]

T = Fmin - k*b0 - (massa*2*b2*(omegamax^2))/1000; [N]
```

Resto[1]	Raggio base [mm]	Diametro bicchierino [mm]	Precarico [N]
0	18.7038	31.4839	482,2200

```
%% dati

h=8; % alzata al netto delle rampe di ripresa gioco [mm]
alpha=240; %[deg]
h1= 0.3; %gioco [mm]
u=270; % max vel d'urto [mm/s]
n=6000; % massimo regime di rotazione motore [rpm]
theta1=38; %[deg]
theta2=42; %[deg]
deltamin=5; %minimo raggio di curvatura del profilo camma
[mm]
Fmin=140; %[N]
k=8.8; % [N/mm]
massa= 0.19; %[kg] massa equipaggio mobile della valvola


%% calcoli

%theta
tq= alpha/4-theta2; % [deg]
thetaR= alpha/4-theta1; % [deg]
thetaS= alpha/4; % [deg]

omega_camma= ( n/2 )* (2*pi/60 ) ; %rad/s
% omegamax = 2pi*n/(60*2) = pi*n/60 [rpm] -->
% omegamax = pi*n/60 * 180/pi=3n [deg/s]
omegamax = 3*n; %[deg/s]



b0=h; %coefficiente legge di alzata [mm]


% yprimo= u/omegamax %[mm/s / deg/s] --> [mm/deg]
omegamax=3*n;
yprimo=u/omegamax; %[mm/deg]

thetaP = -h1/yprimo; %[deg]

format shortE
A=[20*tq^3 12*tq^2 6*tq  0 0; thetaR^5 thetaR^4 thetaR^3 -
(thetaR-thetaS)^4 -(thetaR-thetaS)^2; 5*thetaR^4  4*thetaR^3
3*thetaR^2 -4*(thetaR-thetaS)^3 -2*(thetaR-thetaS);
20*thetaR^3 12*thetaR^2 6*thetaR -12*(thetaR-thetaS)^2 -2;
60*thetaR^2 24*thetaR 6 -24*(thetaR-thetaS) 0];
b=[0; b0-(yprimo*thetaR); -yprimo; 0; 0];
x=A\b;
% disp(x)
a5=x(1);
```

```
a4=x(2);
a3=x(3);
b4=x(4);
b2=x(5);
a0=0;
a1=yprimo;
a2=0;


%% andamento di y in funzione di theta

% lineare da P a O
% pol 5 grado da O a R
% Pol 4 grado da R a S
% speculare da S in poi

% P - O
% m = y' = u/ omegamax
% y = mx+q
% q = 0
% ciclo for che fa variare theta da thetaP a thetaO(0,0)
m= yprimo;
thetaPO = thetaP : 0.1 : 0;
yPO= thetaPO.*m;

% O - R
% y = a5*theta^5 + a4*theta^4 + a3*theta^3 + a2*theta^2 +
a1*theta + a0
thetaOR = 0 : 0.1 : thetaR ;
a0vettore= ones(1,length(thetaOR) ) *a0;
yOR =  a5.*thetaOR.^5 + a4.*thetaOR.^4 + a3.*thetaOR.^3 +
a2.*thetaOR.^2 + a1.*thetaOR + a0vettore ;

% R - S
% y= b4*(thetaRS-thetaS)^4 + b2*(thetaRS-thetaS)^2+ b0
thetaRS = thetaR : 0.1 : thetaS;
b0vettore= ones(1,length(thetaRS) ) *b0;
thetaSvettore= ones(1,length(thetaRS) ) *thetaS;
yRS= b4.*(thetaRS-thetaSvettore).^4 + b2.*(thetaRS-
thetaSvettore).^2+ b0vettore;

thetaPS = [thetaPO thetaOR thetaRS];
yPS= [yPO yOR yRS];
% simmetria
% chiamo U il valore speculare a P

thetaSasse= ones(1,length(thetaPS) ) *thetaS;
distanza = thetaSasse - thetaPS;
thetaUS = thetaPS + distanza.*2;
```

```
yUS=yPS;

figure(1)

plot(thetaPS,yPS)
hold on
plot(thetaUS,yUS)
grid on

% figure(1)
% plot(thetaPO,yPO);
% hold on
% plot(thetaOR,yOR);
% hold on
% plot(thetaRS,yRS);

%% andamento di y' in funzione di theta

% P - O
%y'=dy/dtheta
%yPO= thetaPO.*m
%y'=m
coefvelPO=ones(1,length(thetaPO) )*m;


% O - R
% y = a5*theta^5 + a4*theta^4 + a3*theta^3 + a2*theta^2 +
a1*theta + a0
% y'= 5*a5*theta^4 + 4*a4*theta^3 + 3*a3*theta^2 + 2*a2*theta
+ a1

a1vettore= ones(1,length(thetaOR) ) *a1;
coefvelOR= (5*a5).*thetaOR.^4 + (4*a4).*thetaOR.^3 +
(3*a3).*thetaOR.^2 + (2*a2).*thetaOR + a1vettore;

% R - S
% y= b4*(thetaRS-thetaS)^4 + b2*(thetaRS-thetaS)^2+ b0
% y'= 4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS)
coefvelRS = (4*b4).*((thetaRS-thetaSvettore).^3) +
(2*b2).*(thetaRS-thetaSvettore);
% unisco le 3 parti
coefvelPS= [coefvelPO coefvelOR coefvelRS];

% antisimmetria
% chiamo U il valore speculare a P

coefvelUS=(-1).*coefvelPS;
```

```
figure(2)
plot(thetaPS,coefvelPS)
hold on
plot(thetaUS,coefvelUS)
grid on
% la derivata è antisimmetrica

% plot(thetaPO,coefvelPO)
% hold on
% plot(thetaOR,coefvelOR)
% hold on
% plot(thetaRS,coefvelRS)

%% andamento di y'' in funzione di theta

% derivata di una antisimmetrica è simmetrica


% P - O
%yPO= thetaPO.*m
%y'=m
%y''=0

coefaccPO=zeros(1,length(thetaPO) );

% O - R
% y = a5*theta^5 + a4*theta^4 + a3*theta^3 + a2*theta^2 +
a1*theta + a0
% y'= 5*a5*theta^4 + 4*a4*theta^3 + 3*a3*theta^2 + 2*a2*theta
+ a1
% y''= 20*a5*theta^3 + 12*a4*theta^2 + 6*a3*theta + 2*a2
a2vettore= ones(1,length(thetaOR) ) *a2;
coefaccOR= (20*a5).*thetaOR.^3 + (12*a4).*thetaOR.^2 +
(6*a3).*thetaOR+ 2.*a2vettore;

% R - S
% y= b4*(thetaRS-thetaS)^4 + b2*(thetaRS-thetaS)^2+ b0
% y'= 4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS)
% y''=  12*b4*(thetaRS-thetaS)^2 + 2*b2
b2vettore= ones(1,length(thetaRS) ) *b2;
coefaccRS = (12*b4).*((thetaRS-thetaSvettore).^2) +
2.*b2vettore;



% unisco le 3 parti
coefaccPS= [coefaccPO coefaccOR coefaccRS];

% simmetria
```

```
% chiamo U il valore speculare a P

coefaccUS=coefaccPS;


figure(3)
plot(thetaPS,coefaccPS)
hold on
plot(thetaUS,coefaccUS)
grid on

%% determinare il raggio base della camma

% il minimo raggio di curvatura è pari deltamin

% rho = Rb + h1 + y + y'' >= deltamin
% z(theta)= y + y''
% cerco il minimo di rho --> d( z(theta) ) / dtheta = 0

% ho rischio maggiore di avere un rho basso nella zona che è
più appuntita
% perchè la circonferenza osculatrice è più piccola. La zona
più appuntita
% è quella rappresentata dal polinomio di quarto grado R - S
% chiamo yb yRS per non sovrascrivere le variabili

% yb= b4*(thetaRS-thetaS)^4 + b2*(thetaRS-thetaS)^2+ b0
% yb'= 4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS)
% yb''=  12*b4*(thetaRS-thetaS)^2 + 2*b2
% yb'''= 24*b4*(thetaRS-thetaS)

% d(yb + yb'')/dtheta = yb'+yb'''=
% =4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS) +
24*b4*(thetaRS-thetaS)
% = (thetaRS-thetaS)*[ 4*b4*(thetaRS-thetaS)^2 + 2*b2 + 24*b4
]
% devo imporre la derivata nulla ho due soluzioni:

% 1) thetaRS = thetaS

% 2) 4*b4*(thetaRS-thetaS)^2 + 2*b2 + 24*b4 = 0
% 4*b4*thetaRS^2 + 4*b4*thetaS^2 -8*b4*thetaRS*thetaS + 2*b2
+ 24*b4 = 0
% [ 4*b4*thetaRS^2 ] + [ - 8*b4*thetaS*thetaRS ] +
% +[ b4(4*thetaS^2 + 24)+ 2b2 ]= 0

% thetaRS1,2 = { 8*b4*thetaS +- sqrt{64*(b4^2)*(thetaS^2) -
% - 4*[(b4*(4*thetaS^2 + 24) + 2*b2 )]*4b4} } / (8*b4)
```

```
% delta = 64*(b4^2)*(thetaS^2) - 4*[(b4*(4*thetaS^2 + 24) +
2*b2 )]*4b4 =
% = 64*(b4^2)*(thetaS^2) - 64*(b4^2)*(thetaS^2) - 384*b4^2 -
32*b2*b4 =
% = - 384*b4^2 - 32*b2*b4
% il delta è negativo quindi avrei valori non reali perciò
escludo la
% seconda soluzione

% rhomin = Rb + h1 + z(thetaRS = thetaS) = deltamin
% Metto a sistema:
% 1) Rb =  deltamin - h1 - z(thetaRS = thetaS)
% 2) z = y + y''
% 3) yb=b4*(thetaRS-thetaS)^4 + b2*(thetaRS-thetaS)^2+ b0
% 4) yb''=  12*b4*(thetaRS-thetaS)^2 + 2*b2

% Rb =  deltamin - h1 - yb(thetaS) - yb''(thetaS)  =
% Rb = deltamin - h1 - b0 - 2*b2
% [mm] - [mm] - [mm] - [mm/deg^2]
% devo moltiplicare b2 per *(360/2pi)^2 per avere la stessa
unità di misura

Rb = deltamin - h1 - b0 - 2*b2*( (360/(2*pi))^2 ); % [mm]

%%  disegnare il profilo della camma in scala 2:1

% conviene utilizzare le cordinate polari OE , theta + delta
% OE^2 = OD^2 + OC^2
% OC = y'
% OD = Rb + h1
% tan(delta) = OC / OD

% devo tenere conto di 3 zone:

% 1) U - P (198°)
% RAGGIO BASE : OD = Rb ; OC = 0;
figure(4)
OD1=Rb;
OC1=0;
OE1 = sqrt(OD1^2+OC1^2); %OE^2 = OD^2 + OC^2
delta1=atan(OC1/OD1); % arctan(0/Rb)
%thetaU=thetaS+(thetaS-thetaR)+(thetaR-thetaP);
%thetaN=2*thetaS;%thetaS+(thetaS-thetaR)+(thetaR-
thetaP)+thetaP=2*thetaS;
thetaU=thetaS+(thetaS-thetaR)+(thetaR-thetaP);
thetaPnormalizzato=thetaP+360;
thetaUP=thetaU : 0.1 : thetaPnormalizzato;
theta1=thetaUP*pi/180; %[rad]
OE1vettore= ones(1,length(theta1) )*OE1;
```

```
deltalvettore= ones(1,length(theta1) )*delta1;
angolo1=theta1+delta1;
% 2) P - O + N - U (18°+18°)
% TRATTO DI RECUPERO GIOCO :
% OD = Rb + h1 + y0'
% y0'= OC = costante

% P - O
Rbvettore= ones(1,length(thetaPO) ) *Rb;
h1vettore= ones(1,length(thetaPO) ) *h1;
OD2a= Rbvettore + h1vettore + coefvelPO.*thetaPO;
OC2a= coefvelPO*180/pi;
delta2a=atan(OC2a./OD2a);
OE2a = sqrt(OD2a.^2+OC2a.^2);
theta2a = thetaPO*pi/180; %[rad]
angolo2a= theta2a + delta2a;

% N - U

%thetaT=2*(thetaS-thetaR) + thetaR;
%thetaTvettore=ones(1,length(thetaOR) ).*thetaT;
%thetaTN = thetaOR  + thetaTvettore;

thetaO=0;
thetaN= thetaS +(thetaS - thetaO);
thetaNU = thetaPO + (thetaN - thetaP);% non va normalizzato
%coefvelTN=(-1).*coefvelOR;
coefvelNU=(-1).*fliplr(coefvelPO); % non va normalizzato
OD2b= Rbvettore + h1vettore + coefvelNU.*thetaNU;
OC2b= coefvelNU*180/pi;
OE2b = fliplr(OE2a);
delta2b=atan(OC2b./OD2b);
theta2b = thetaNU*pi/180; %[rad]
angolo2b= theta2b + delta2b;

%  continuità
OE12= [OE2b OE1vettore OE2a];
angolo12 = [angolo2b angolo1 angolo2a];

% 3) O - N (126°)
% TRATTO ATTIVO :
% OD = Rb + h1 + y
% OC = y'

% yST=fliplr(yRS);
% yRT=[yRS yST];
yOS=[yOR yRS(2:end)];
ySN=fliplr(yOS);
yON=[yOS ySN(2:end)];
```

```
% coefvelST=fliplr(coefvelRS).*(-1);
% coefvelRT= [coefvelRS coefvelST];
coefvelOS=[coefvelOR coefvelRS(2:end)];
coefvelSN=fliplr(coefvelOS).*(-1);
coefvelON= [ coefvelOS coefvelSN(2:end)];
Rbvettore3= ones(1,length(yON) ).*Rb;
h1vettore3= ones(1,length(yON) ).*h1;
% thetaST= thetaRS + (thetaS - thetaR);
% thetaRT=[thetaRS thetaST];
thetaON=thetaO:0.1:thetaN;
OD3= Rbvettore3 + h1vettore3 + yON;
OC3= coefvelON*180/pi;
delta3=atan(OC3./OD3);
OE3 = sqrt(OD3.^2+OC3.^2);
theta3 = thetaON*pi/180; %[rad]
angolo3= theta3 + delta3;

% continuità
OE= [OE3 OE12];
angolo = [angolo3 angolo12];

% figure(4)
% polarplot(angolo1,2*OE1vettore,'LineWidth',2);
% hold on
% polarplot(angolo2a,2*OE2a,'LineWidth',1);
% polarplot(angolo12,2*OE12,'LineWidth',2);
% hold on
% polarplot(angolo2b,2*OE2b,'LineWidth',1);
% hold on
% polarplot(angolo3,2*OE3,'LineWidth',3);

figure(4)
polarplot(angolo,2*OE,'LineWidth',3);

%% bicchierino

% determino la massima distanza dall'asse valvola del
contatto
% camma-bicchierino

%devo avere sempre contatto fra camma e bicchierino quindi il
bicchierino
%deve avere un ampiezza almeno uguale a y'max

OCmax=max([OC1 OC2a OC2b OC3]); %[mm]

% ricorda : OC = y'=dy/dtheta quindi devo correggere l'unità
di misura se
% voglio una distanza in mm
```

```
% determino il diametro minimo d del bicchierino essendo pari
a d/2 lo
% spessore della camma

%d/2= y'max / cos(30)
%y'max= OCmax

d= 2 * OCmax / cos(pi/6);

%% precarico molla

% F = Fel + Fin  + T;
% T = Precarico
% Fel = Felastica
% Fel = - ky
% Fin = Finerziale
% Fin = - m*d^2y/dt^2 = -m*coefacc*omega^2

% Cerco la Fmin Fmin =  (Fel + Fin)min  + T;
% il minimo si raggiunge sul polinomio yb (zona R - S )
% (Fel + Fin)min = - (ky + m*coefacc*omega^2 )
% [N/mm] * [mm] + [kg * mm / s^2 * 1/1000]

% devo trovare il minimo di - (k*yRS + m*coefaccRS*omegamax^2
)
% d( (Fel + Fin) ) / dtheta = 0
% d ( (k*yRS + m*coefaccRS*omegamax^2 ) ) = 0
% k*yb' + m* yb''' * omegamax^2 = 0

% ricavate precedentemente (zona R - S) :
% yb= b4*(thetaRS-thetaS)^4 + b2*(thetaRS-thetaS)^2+ b0
% yb'= 4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS)
% yb''=  12*b4*(thetaRS-thetaS)^2 + 2*b2
% yb'''= 24*b4*(thetaRS-thetaS)

% k*(4*b4*(thetaRS-thetaS)^3 + 2*b2*(thetaRS-thetaS) ) +
% + m*omegamax^2*24*b4*(thetaRS-thetaS)= 0

%(thetaRS-thetaS)*[4k*b4*(thetaRS-thetaS)^2 +2k*b2 +
24m*omegamax^2*b4] = 0
% ho due soluzioni.( thetaRS = theta sono letteralmente la
stessa cosa )

% 1) theta = thetaS

% 2) 4k*b4*(theta-thetaS)^2 +2k*b2 + 24m*omegamax^2*b4 = 0
```

```
% 4k*b4*theta^2 + 4k*b4*thetaS^2 - 8k*b4*theta*thetaS + 2k*b2
+ 24m*omegamax^2*b4 = 0
% 4k*b4*(theta^2) - 8k*b4*thetaS*(theta) + (4k*b4*thetaS^2 +
2k*b2 + 24m*omegamax^2*b4 ) = 0

% theta1,2=  8k*b4*thetaS +-
% +- sqrt (64*k^2*b4^2*thetaS^2 -
4*(4k*b4*thetaS^2+2k*b2+24m*omegamax^2*b4 )*(4k*b4))

% delta = 64*k^2*b4^2*thetaS^2 -
4*(4k*b4*thetaS^2+2k*b2+24m*omegamax^2*b4)*(4k*b4) =
% = 64*k^2*b4^2*thetaS^2 - 4^3*b4^2*k^2*thetaS^2 -
4^2*2*k^2*b2*b4 -
% - 4^2*24*m*omegamax^2*b4^2 =
% =  - 32*k^2*b2*b4 - 384*m*omegamax^2*b4^2 < 0
% delta < 0 --> le soluzioni non sono reali
% utilizzo perciò la 1 soluzione

% 1) theta = thetaS
% T = Fmin - k*yb(thetaS) - m*coefaccRS(thetaS)*omegamax^2

% T = Fmin - k*bo - m*2b2*omegamax^2

% Fmin = [N]
% k = [N /mm]
% bo = [mm]
% omegamax=3*n;  [mm/deg^2]
% b2 = [deg^2/s^2]
% m= massa equipaggio mobile della valvola [kg]

% [N] - [N /mm] * [mm] - [kg]*[deg^2/s^2]*[mm/deg^2]
% [N] - [N ] - [kg*mm/s^2]
% [N] - [N ] - [N*1/1000]

T = Fmin - k*b0 - (massa*2*b2*(omegamax^2))/1000;
```

5 Riduttore a due stadi

In Fig. 8 è rappresentato un riduttore coassiale a due stadi le cui ruote dentate sono cilindriche a denti dritti.
Volendo garantire la coassialità tra albero di ingresso (su cui è calettata la ruota 1) e quello di uscita (su cui è montata la ruota 4), i dati impongono di operare una correzione.
Si sceglie di intervenire sul secondo stadio, per il quale si chiede di determinare:

1. l'angolo di pressione di lavoro $\alpha_{3,4}$ (espresso in gradi);
2. la somma dei coefficienti di spostamento $x_3 + x_4$;
3. la somma degli spostamenti di profilo $v_3 + v_4$ (espressa in mm).

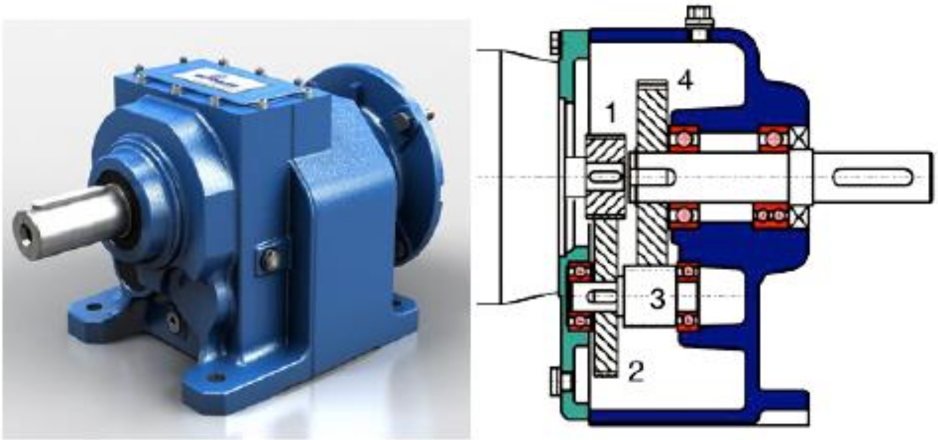


Figura 8: *Riduttore coassiale a due stadi.*

DATI

u è il resto della divisione per 4 del numero di matricola.

Modulo del primo stadio $m_{1,2}$	$= 2.5\ mm$
Modulo del secondo stadio $m_{3,4}$	$= 3\ mm$
Numero di denti delle ruote:	
Z_1	$= 50$
Z_2	$= 160\ +u$
Z_3	$= 59\ +u$
Z_4	$= 118\ -u$
Angolo di pressione di taglio α	$= 20°$

```
%% RAGGI PRIMITIVI DELLE RUOTE DENTATE
Calcolo i raggi primitivi delle ruote conoscendo il modulo e
il numero dei denti
La formula utilizzata è:
Raggio primitivo = modulo * numero di denti / 2
Modulo del primo stadio: M1_2 = 2.5; [mm]
Modulo del secondo stadio: M3_4 = 3; [mm]
R1=M1_2*Z1/2; [mm]
R2=M1_2*Z2/2; [mm]
```

```
R3=M3_4*Z3/2; [mm]
R4=M3_4*Z4/2; [mm]

%% INTERASSI DI TAGLIO
```

Noti i raggi primitivi calcolo gli interassi di taglio

Interasse di taglio del 1° stadio: a1_2= R1+R2; [mm]
Interasse di taglio del 2° stadio: a3_4= R3+R4; [mm]

```
%% COASSIALITA' TRA I DUE ALBERI
```

Decido di correggere il 2° stadio e per garantire la coassialità impongo: a3_4_primo = a1_2

Dove a3_4_primo è l'interasse di lavoro

```
a3_4_primo = a1_2; [mm]


%% ANGOLI DI PRESSIONE
```

Ricordando che il raggio della circonferenza di base non varia cerco l'angolo di pressione.

Rho= R*cos(alpha) = R_primo*cos(alpha_primo)

R_primo= R*cos(alpha)/ cos(alpha_primo)

Ma essendo l'interasse la somma dei raggi primitivi posso scrivere che:

% a3_4_primo = a3_4 * cos(alpha) / cos (alpha3_4)

Ricavo quindi alpha3_4

```
alpha3_4 = acos( a3_4 * cos(alpha) /  a3_4_primo ); [rad]


%% COEFFICIENTI DI SPOSTAMENTO
```

Creo una function che mi calcola la funzione evolvente
Involute(alpha) = tan(alpha) - alpha

Teoria: involute(alpha_primo)= involute(alpha0) + 2*tan(alpha0) * (x1+x2) / (z1+z2)

```
Ricavo poi X3+X4

2*tan(alpha) * (X3+X4) + (Z3+Z4)*(involute(alpha) -
involute(alpha3_4) ) = 0

Somma_X3_X4= (Z3+Z4)*( - involute(alpha) + involute(alpha3_4)
) / (2*tan(alpha) ); [mm]


%% SPOSTAMENTI DI PROFILO

Ricordando che v= m*x

Somma_v3_v4 = Somma_X3_X4 * M3_4 [mm]
```

Resto[1]	**Angolo di pressione di lavoro $\alpha_{3,4}$ [*deg*]**	**Somma coefficienti di spostamento $x_3 + x_4$**	**Somma spostamenti di profilo $v_3 + v_4$ [*mm*]**
0	18.1154	-0.9555	-2.8666

```
u=0; % u è il resto della divisione per 4 del numero di
matricola

%% DATI

M1_2 = 2.5; % [mm] Modulo del primo stadio
M3_4 = 3; % [mm] Modulo del secondo stadio
% Numero di denti delle ruote:
Z1 = 50;
Z2 = 160 +u ;
Z3 = 59 +u ;
Z4 = 118 - u;
alpha = 20*pi/180; % [rad] Angolo di pressione di taglio

%% RAGGI PRIMITIVI DELLE RUOTE DENTATE

R1=M1_2*Z1/2; %[mm]
R2=M1_2*Z2/2; %[mm]
R3=M3_4*Z3/2; %[mm]
R4=M3_4*Z4/2; %[mm]

%% INTERASSI DI TAGLIO

% a= interassee di taglio

a1_2= R1+R2; %[mm] 1 stadio
a3_4= R3+R4; %[mm] 2 stadio

%% COASSIALITA' TRA I DUE ALBERI

% correggo il 2° stadio

% condizione di coassialità : a3_4_primo = a1_2
% (a3_4_primo va calcolato dopo la correzione)

a3_4_primo = a1_2; % a_primo=interasse di lavoro

%% ANGOLI DI PRESSIONE

% a3_4_primo = a3_4 * cos(alpha) / cos (alpha3_4)
alpha3_4 = acos( a3_4 * cos(alpha) /  a3_4_primo ); % [rad]

%% COEFFICIENTI DI SPOSTAMENTO

%cerco la somma dei coefficienti di spostamento X3+X4

% teoria: involute(alpha_primo)= involute(alpha0) +
2*tan(alpha0) * (x1+x2) / (z1+z2)
```

```
% 2*tan(alpha) * (X3+X4) + (Z3+Z4)*(involute(alpha) -
involute(alpha3_4) ) = 0

Somma_X3_X4= (Z3+Z4)*( - involute(alpha) + involute(alpha3_4)
) / (2*tan(alpha) );

%% SPOSTAMENTI DI PROFILO

%cerco la somma degli spostamenti di profilo v3+v4 [mm]
% v= m*x

Somma_v3_v4 = Somma_X3_X4 * M3_4
```

9 Frequenza propria di una colonna con serbatoio elevato

Determinare la prima frequenza propria di vibrazione flessionale della colonna con serbatoio elevato mostrata in Fig. 19, supponendo che la sezione tubolare della colonna sia costante.
Si esprima il risultato in Hz impiegando almeno cinque cifre significative.

Simboli:
D = diametro esterno della colonna
d = diametro interno della colonna
l = lunghezza della colonna
E = modulo di elasticità del materiale della colonna
Q = peso del serbatoio
ρ = massa volumica del materiale della colonna

Figura 19: *Serbatoio elevato.*

DATI

uv ultime due cifre del numero di matricola
(matricola = 0000####uv).

$D = 3 + u/10$ $[m]$
$\rho = 2400 + v^2 + u$ $[kg/m^3]$
$d = 2.45 + v/30$ $[m]$
$E = 2.8 \times 10^{10}$ $[N/m^2]$
$l = 90 + u^2/5 - v$ $[m]$
$Q = (2.7 + u^2/100 + u \times v/50) \times 10^6$ $[N]$

%% SEZIONE DELLA TRAVE

Calcolo il momento di inerzia, la rigidezza flessionale e la sezione della colonna

Approssimo il mio sistema ad una trave incastrata con carico all'estremità

momento di inerzia trave incastrata I=pi*(D^4-d^4)/64 [mm]

P= carico all'estremità

equazione linea elastica :y(x)= P*x^2 / (6*E*I) * (3*l-x);

ymax =y(x=l) = P*(l^3) / 3*E*I;

k= rigidezza flessionale

K si trova imponendo P=1--> k= 1/ymax

k = 1 / ((l^3) / 3*E*I) [N/m]

controllo : 1 / { [m^3] / ([N/m^2]*[m^4]) } = [N/m]

sezione della trave: S=pi * (D^2-d^2)/4; [m^2]

%% Metodo energetico di Rayleigh Tmax= Vmax

Cerco la prima pulsazione utilizzando il metodo energetico di Rayleigh

funzione vibratoria : v(t,x)= f(t)*y(x);
f(t)= funzione del tempo
y(x) = funzione della cordinaata spaziale

A= ampiezza
w1 = approssimazione 1 pulsazione
Considero : f(t)= A*cos(w1*t)
df(t)/dt= -w1*A*sin(w1*t)

energia cinetica: T= Tcolonna + Tserbatoio

Tcolonna = Tc

$$\mathrm{Tc} = ½ * \int_0^m \left(\frac{\partial \mathrm{v}}{\partial \mathrm{t}}\right)^2 dm = ½ * \int_0^m \left(\frac{\partial \mathrm{v}}{\partial \mathrm{t}}\right)^2 \mathrm{rho} * \mathrm{S} * dx = ½ * \mathrm{rho} * \mathrm{S} * \int_0^m \left(\frac{\partial \mathrm{v}}{\partial \mathrm{t}}\right)^2 * dx$$

$$\mathrm{Tc} = ½ * \mathrm{rho} * \mathrm{S} * \int_0^m y(x)^2 * \left(\frac{\mathrm{df}}{\mathrm{dt}}\right)^2 * dx = ½ * \mathrm{rho} * \mathrm{S} * \left(\frac{\mathrm{df}}{\mathrm{dt}}\right)^2 * \int_0^m y(x)^2 * dx =$$

$$\mathrm{Tc} = ½ * \mathrm{rho} * \mathrm{S} * \left(\frac{\mathrm{df}}{\mathrm{dt}}\right)^2 * \int_0^m \frac{ymax^2}{4} * \left(\frac{9\mathrm{x}^4}{\mathrm{l}^4} + \frac{\mathrm{x}^6}{\mathrm{l}^6} - 6 * \frac{\mathrm{x}^5}{\mathrm{l}^5}\right) * dx =$$

$$\mathrm{Tc} = ½ * \mathrm{rho} * \mathrm{S} * \left(\frac{\mathrm{df}}{\mathrm{dt}}\right)^2 * \frac{ymax^2}{4} * \left(\frac{9}{5} l + \frac{1}{7} l - l\right)$$

$$Tc = \frac{1}{8} * \mathrm{rho} * \mathrm{S} * \left(\frac{\mathrm{df}}{\mathrm{dt}}\right)^2 * ymax^2 * \left(\frac{33}{35} l\right) \quad [\mathrm{J}]$$

Tserbatoio = Ts

$$Ts = ½ * M * \left(\left(\frac{\partial v}{\partial t}\right)^2 |x = l \right) = \frac{1}{2} * \frac{Q}{g} * \left(\frac{df}{dt}\right)^2 * ymax^2$$ [J]

T = (((df(t)/dt)^2) * ymax^2) * (1/8 * rho*S*l *33/35 + +1/2 Q/g) [J]

V= 1/2 * k * (v(t,x=l)^2 =1/2 * 3*E*I / (l^3) *f(t)^2 *ymax^2 [J]

Rayleigh Tmax = Vmax

Tmax = w1^2 * A^2* max((sin(w1*t))^2) * ymax^2 *(1/8* *rho*S*l*33/35 + 1/2 Q/g)

max((sin(w1*t))^2) =1

Vmax= 1/2 * k *A^2 *max((cos(w1*t))^2) *ymax^2

max((cos(w1*t))^2) = 1

Tmax= Vmax

w1^2 * A^2 * ymax^2 *(1/8 *rho*S*l*33/35 + 1/2 Q/g) = 1/2 * * k *A^2 *ymax^2

w1^2 = 1/2 * k / (33/280 *rho*S*l + 1/2 Q/g)

$$w1 = \sqrt{\frac{\frac{\frac{3}{2}E*I}{l^3}}{\frac{33}{280}*rho*S*l+\frac{Q}{2g}}}$$ [rad/s]

prima frequenza del sistema : f1= w1/(2*pi) [1/s]

Ultime due cifre del numero di matricola		Prima frequenza propria [*Hz*]
***u*=8**	***v*=4**	0.1539

```
u=8;
v=4;
%% DATI

D = 3 + u/10; % [m]  diametro esterno della colonna
rho = 2400 + v^2 + u; %[kg/m3] massa volumica del materiale
della colonna
d = 2.45 + v/30; % [m]  diametro interno della colonna
E = 2.8 * 10^10 ; % [N/m2]  modulo di elasticità del
materiale della colonna
l = 90 + (u^2) /5 - v; % [m]  lunghezza della colonna
Q = (2.7 + (u^2) /100 + u * v/50) * 10^6; %[N]  peso del
serbatoio
g=9.81; %[m/s^2]
%% SEZIONE DELLA TRAVE

%la prima frequenza di vibrazione si studia ipotizzando la
prima forma
%modale alla linea elastica. Il mio sistema è approssimabile
a una trave
%incastrata con carico all'estremità

x=[0:1:l]; % posizione nella trave
I=pi*(D^4-d^4)/64; %momento di inerzia trave incastrata
% P= carico all'estremità
% y(x)=P*x^2 / (6*E*I) * (3*l-x); %equazione linea elastica
%ymax =y(x=l) =P*(l^3) / 3*E*I;


% k= rigidezza flessionale .
% K si trova imponendo P=1--> k= 1/ymax
k = 1 / ( (l^3) / 3*E*I );

% S= sezione della trave
S=pi * (D^2-d^2)/4;
%% Metodo energetico di Rayleigh Tmax= Vmax

% funzione vibratoria
% v(t,x)= f(t)*y(x);
% f(t)= funzione del tempo
% y(x) = funzione della cordinaata spaziale

% A= ampiezza
% w1 = approssimazione 1 pulsazione
%f(t)= A*cos(w1*t)
%df(t)/dt= -w1*A*sin(w1*t)

% T= energia cinetica. T= Tcolonna + Tserbatoio
```

```
% Tc=1/8*rho*S*( (df(t)/dt)^2 ) * ymax^2 *33/35 *l
% Ts=1/2 * Q/g * ( ( (df(t)/dt)^2 ) * ymax^2 )

% T = ( ( (df(t)/dt)^2 ) * yma x^2 ) * (1/8 * rho*S*l *33/35
+ 1/2 Q/g )

% V= 1/2 * k * (v(t,x=l)^2 =1/2 * 3*E*I / (l^3 ) *f(t)^2
*ymax^2

% Tmax = w1^2 * A^2* max( (sin(w1*t))^2 ) * ymax^2 *(1/8
*rho*S*l*33/35 + 1/2 Q/g )
% max( (sin(w1*t))^2 ) =1

% Vmax= 1/2 * k  *A^2 *max( (cos(w1*t))^2 ) *ymax^2
% max( (cos(w1*t))^2 )

% Tmax= Vmax
% w1^2 * A^2 * ymax^2 *(1/8 *rho*S*l*33/35 + 1/2 Q/g ) = 1/2
* k  *A^2 *ymax^2
% w1^2  = 1/2 * k / (33/280 *rho*S*l + 1/2 Q/g )
w1= sqrt( ( (3/2 *E*I ) / (l^3 ) )  / (33/280 *rho*S*l + Q
/(2*g) ) ); % rad/s

% f1 = prima frequenza del sistema
f1= w1/( 2*pi) % 1/s
```

12 Vibrazioni torsionali di un motore marino

In Fig. 24(a) è rappresentato lo schema di un motore marino connesso all'elica mediante un riduttore ad ingranaggi ad uno stadio. Noti i momenti di inerzia del volano, del motore, delle ruote dentate, dell'elica e le dimensioni degli alberi, trovare le frequenze naturali e i modi di vibrare torsionali del sistema.

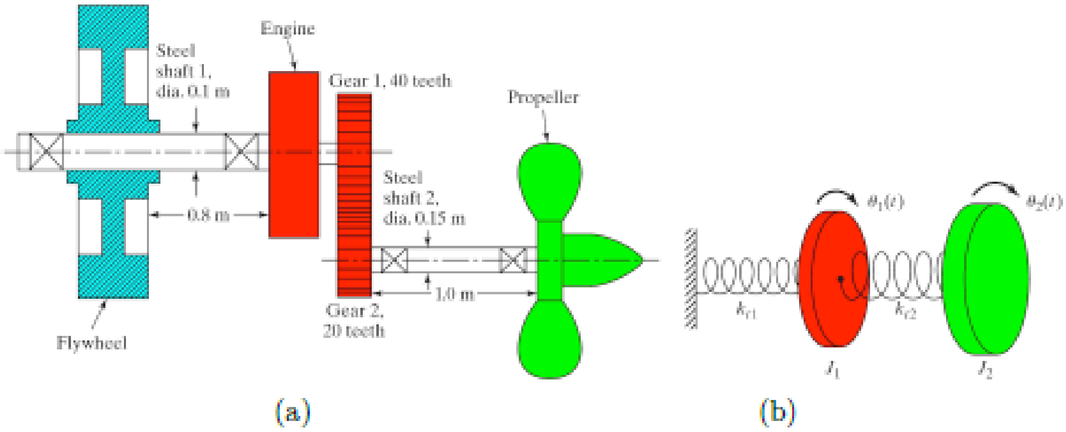


Figura 24: a) *Motore marino;* b) *Modello.*

Si trascuri l'inerzia degli alberi e si esprima il risultato utilizzando almeno cinque cifre significative. Inoltre:

1. si esprimano le frequenze naturali in Hz;
2. indicata con θ la rotazione dell'asse motore e con φ la rotazione dell'asse dell'elica, esprimere i modi di vibrare nella seguente forma:

$$r_1 = \left\{\frac{\Phi_2}{\Theta_1}\right\}_1 \qquad r_2 = \left\{\frac{\Phi_2}{\Theta_1}\right\}_2$$

DATI

u ultima cifra del numero di matricola (matricola = 0000#####u).

J_v momento di inerzia del volano	$= 35000\ [kg \cdot m^2]$
J_m momento di inerzia del motore	$= 1000 - 5 \times u^2\ [kg \cdot m^2]$
J_1 momento di inerzia della ruota 1	$= 250 - u\ [kg \cdot m^2]$
J_2 momento di inerzia della ruota 2	$= 150 + 2 \times u\ [kg \cdot m^2]$
J_e momento di inerzia dell'elica	$= 2000 + 20 \times u^2\ [kg \cdot m^2]$
G modulo di elasticità tangenziale acciaio	$= 8 \times 10^{10}\ [N/m^2]$

%% MODELLO A PARAMETRI CONCENTRATI

Nel modello a parametri concentrati studio solo inerzie perfettamente rigide e molle perfettamente elastiche

1° asse uso cordinata theta

thetav= gdl rotazione volano
thetam= gdl rotazione motore
k1= molla torsionale con rigidezza uguale a quella dell'albero motore

2° asse uso cordinata phi

k2= molla torsionale con rigidezza uguale a quella dell'albero elica
phi_2= posizione angolare dell'elica
phiZ_2=posizione angolare dell'inerzia Jz_2

phiZ_2 dipende da thetam quindi non è un GDL nel mio modello

GDL:

thetav= gdl rotazione volano
thetam= gdl rotazione motore
phi_2= posizione angolare dell'elica

Calcoli:

coppia torcente unitaria per trovare K torsionale:
Coppia_torcente=1[N m]

momento di inerzia polare di sezione dell'albero motore: Ip_1= pi*dm^4/32 [m^4]

momento di inerzia polare di sezione dell'albero elica:
Ip_2= pi*de^4/32 [m^4]

deformazione angolare causata da M=1:
deltaTHETA_1= (Coppia_torcente*lm) / (G*Ip_1)
([N*m]*[m]) / ([N/m^2] * [m^4])= [N*m^2 / N*m^2] = []

K torsionale relativa all'albero motore:
K1= 1/deltaTHETA_1 [n*m]

deformazione angolare causata da M:
deltaTHETA_2= (Coppia_torcente*le) / (G*Ip_2)

K torsionale relativa all'albero elica:
K2= 1/deltaTHETA_2 [N*m]

```
phi_punto = derivata rispetto al tempo di phi
theta_punto = derivata rispetto al tempo di theta

Rapporto di trasmissione:
tau = z1/z2 = omega2/omega1
omega2 = phi_punto = vel angolare a valle
omega1 = theta_punto = vel angolare a monte

%% IPOTESI

1)
     Ipotizzo l'albero tra motore e ruota 1 infinitamente rigido
     torsionalmente dato che è di lunghezza << lm e le .
     grazie a questa ipotesi posso affermare che Jz_1 e Jm siano
     rigidamente collegati

2)
     Stesso ragionamento per Jz_2 e Je anche se ho un albero
     diverso su un asse diverso --> Jz_2 e Je sono rigidamente
     collegati

     Ho 3 gdl --> mi aspetto quindi 3 pulsazioni naturali
     Osservando che il sistema non è collegato a telaio mi aspetto
     1 Moto rigido e 2 moti vibratori ( w1=0 ; w2=\= 0 ; w3 =\=0 )

3)
     Volano perfetto --> Inerzia infinita --> theta 1° sezione =
     costante
     Studio un modello equivalente con un incastro in
     corrispondenza di Jv perchè Jv  molto maggiore rispetto alle
     altre inerzie ( Jv >> Je > Jm )→ Avrò quindi solo 2
     pulsazioni naturali ma molto simili a quelle del modello senza
     volano perfetto

%% MODELLO A 2 GDL

GDL:

thetam= gdl rotazione motore
phi_2= posizione angolare dell'elica



%% RIDUZIONE AL 1° ASSE

Voglio ridurre tutto al primo asse, perciò, faccio delle
equivalenze dinamiche per cercare le variabili dinamiche

nuove variabili:
Jz_2_primo
Je_primo
K2_primo
```

Energia cinetica:

T (Jz_2) : 1/2 *Jz_2 * phi_punto^2 = 1/2* Jz_2_primo *
*theta_punto^2

--> Jz_2_primo = Jz_2 * (phi_punto^2 / theta_punto^2) = Jz_2 *
*tau^2

Jz_2_primo = Jz_2 * tau^2 [kg * m^2]

T (Je) :
Je_primo = Je * tau^2 [kg * m^2]

Energia potenziale :

1/2 K2* (delta_phi)^2 = 1/2 K2_primo* (delta_theta)^2

--> K2_primo = K2*tau^2 [N*m]

%% EQUAZIONI DI EQUILIBRIO SISTEMA RIDOTTO

Theta2 = theta elica

1) (Jm + Jz_1 + Jz_2_primo) * thetam_2punti + K1*thetam +
+K2_primo*thetam - K2_primo*thetaelica =0

2) Je_primo * thetaelica_2punti + K2_primo*thetaelica-
-K2_primo*thetam = 0

M=

Jm + Jz_1 + Jz_2_primo	0
0	Je_primo

K=

K1 + K2_primo	-K2_primo
-K2_primo	K2_primo

%% RISULTATI

Uso eig per trovare la matrice dinamica e la matrice degli
autovettori

```
[V,D]=eig(K,M);
% D = matrice dinamica diagonale
% V= matrice avente come colonne gli autovettori
```

```
Ricordando che le frequenze vanno ordinata in ordine crescente
estraggo le pulsazioni naturali dalla matrice dinamica diagonale
D ha come elementi le pulsazioni naturali al quadrato

Una volta ricavate w1 e w2 calcolo le frequenze  naturali:
f1= w1/(2*pi) [s^-1]
f2= w2/(2*pi) [s^-1]


Calcolo infine i modi di vibrare:

r1 = { PHI2 / THETA1 }|1 = (tau*phi2 /theta1 )|1
--> r1 = tau * theta21/theta11

r2 = { PHI2 / THETA1 }|2 = (tau*phi2 /theta1 )|2
--> r2 = tau * theta22/theta12
```

Ultima cifra del numero di matricola	**Prima frequenza propria [*Hz*]**	**Seconda frequenza propria [*Hz*]**	**Rapporto $r_1 = [\Phi_2/\Theta_1]_1$**	**Rapporto $r_2 = [\Phi_2/\Theta_1]_2$**
***u*=4**	1.4666	16.7078	2.1042	-0.3683

```
%% DATI

u = 4; % ultima cifra del numero di matricola

Jv= 35000; % [kg · m^2] momento di inerzia del volano
Jm = 1000 - 5 * u^2; %momento di inerzia del motore [kg ·
m^2]
Jz_1 = 250 - u;% [kg · m^2] momento di inerzia della ruota 1
Jz_2 = 150 + 2 * u;% [kg · m^2] momento di inerzia della
ruota  2
Je = 2000 + 20 * u^2;%[kg · m2] momento di inerzia dell’elica
G = 8 * 10^10 ;%[N/m^2] modulo di elasticità tangenziale
acciaio
lm = 0.8; %[m] lunghezza albero motore
le = 1; %[m] lunghezza albero elica
dm =0.1; %[m] diametro albero motore
de =0.15; %[m] diametro albero elica
z1= 40; % denti_ruota1
z2= 20; % denti_ruota2

%% MODELLO A PARAMETRI CONCENTRATI

% 1° asse uso cordinata theta

% thetav= gdl rotazione volano;
```

```
% thetam= gdl rotazione motore;
% k1= molla torsionale con rigidezza uguale a quella
dell'albero motore

% 2° asse uso cordinata phi

% k2= molla torsionale con rigidezza uguale a quella
dell'albero elica
% phi_2= posizione angolare dell'elica
% phiZ_2=posizione angolare dell'inerzia Jz_2

% GDL

% thetav= gdl rotazione volano;
% thetam= gdl rotazione motore;
% phi_2= posizione angolare dell'elica

% phiZ_2 non è un gdl perchè dipende da thetam
% tau = z1/z2 = omega2/omega1
% omega2 = phi_punto vel angolare a valle
% omega1 = theta_punto vel angolare a monte

Coppia_torcente=1; % coppia torcente unitaria per trovare K
torsionale
Ip_1= pi*dm^4/32; %momento di inerzia polare di sezione
dell'albero motore
Ip_2= pi*de^4/32; %momento di inerzia polare di sezione
dell'albero elica

deltaTHETA_1= ( Coppia_torcente*lm ) / (G*Ip_1); %
deformazione angolare causata da M
K1= 1/deltaTHETA_1; %K torsionale relativa all'albero motore

deltaTHETA_2= ( Coppia_torcente*le ) / (G*Ip_2); %
deformazione angolare causata da M
K2= 1/deltaTHETA_2; %K torsionale relativa all'albero elica

tau= z1/z2;
%% IPOTESI

% 1)
% ipotizzo l'albero tra motore e ruota 1 infinitamente rigido
% torsionalmente dato che è di lunghezza << lm e le .
% grazie a questa ipotesi posso affermare che Jz_1 e Jm siano
rigidamente collegati

% 2)
% Stesso ragionamento per Jz_2 e Je anche se ho un albero
diverso su un
```

```
% asse diverso --> Jz_2 e Je sono rigidamente collegati

% Ho 3 gdl --> mi aspetto quindi 3 pulsazioni naturali
% Osservando che il sistema non è collegato a telaio mi
aspetto 1 Moto
% rigido e 2 moti vibratori ( w1=0 ; w2=\= 0 ; w3 =\=0 )

% 3)
% Volano perfetto --> Inerzia infinita --> theta 1° sezione =
costante
% Studio un modello equivalente con un incastro in
corrispondenza di Jv
% perchè Jv  molto maggiore rispetto alle altre inerzie ( Jv
>> Je > Jm )
% avrò quindi solo 2 pulsazioni naturali ma molto simili a
quelle del
% modello senza volano perfetto

%% MODELLO A 2 GDL

% GDL
% thetam= gdl rotazione motore
% phi_2= posizione angolare dell'elica

%% RIDUZIONE AL 1° ASSE

% _primo = riportato all'asse motore

% nuove variabili
% Jz_2_primo
% Je_primo
% K2_primo

%faccio delle equivalenze dinamiche per cercare le variabili
dinamiche

% _punto = derivata prima
% _2punti = derivata seconda

% Energia cinetica:

% T (Jz_2) : 1/2 *Jz_2 * phi_punto^2 = 1/2* Jz_2_primo
*theta_punto^2
% --> Jz_2_primo= Jz_2 * ( phi_punto^2 / theta_punto^2 ) =
Jz_2 * tau^2

Jz_2_primo= Jz_2 * tau^2; % [kg * m^2]

% T (Je) :
```

```
Je_primo = Je * tau^2; % [kg * m^2]

% energia potenziale :
% 1/2 K2* (delta_phi)^2 = 1/2 K2_primo* (delta_theta)^2

K2_primo= K2*tau^2; % [N*m]

%% EQUAZIONI DI EQUILIBRIO SISTEMA RIDOTTO

%theta2=theta elica

% 1) ( Jm + Jz_1 + Jz_2_primo) * thetam_2punti + K1*thetam
+K2_primo*thetam - K2_primo*theta2 =0

% 2) Je_primo * theta2_2punti + K2_primo*theta2 -
K2_primo*thetam = 0

M = [ ( Jm + Jz_1 + Jz_2_primo) , 0 ; 0 Je_primo ];
K = [ (K1 + K2_primo) , -K2_primo ; -K2_primo , K2_primo];

%% RISULTATI

[V,D]=eig(K,M);
% D = matrice dinamica diagonale
% V= matrice avente come colonne gli autovettori
theta11=V(1,1);
theta12=V(1,2);
theta21=V(2,1);
theta22=V(2,2);

w1= sqrt(D(1,1) ); % 1° pulsazione naturale
w2= sqrt(D(2,2) ); % 2° pulsazione naturale

% frequenze naturali
f1= w1/(2*pi) %[s^-1]
f2= w2/(2*pi) %[s^-1]

%modi di vibrare

% r1 = { PHI2 / THETA1 }|1 = (tau*phi2 /theta1 )|1
r1 = tau * theta21/theta11

% r2 = { PHI2 / THETA1 }|2 = (tau*phi2 /theta1 )|2
r2 = tau * theta22/theta12
```

13 Modifiche strutturali

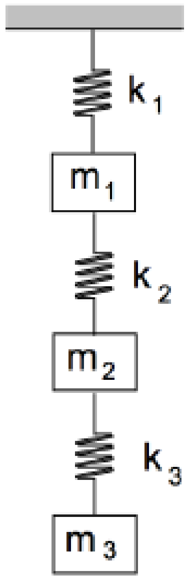
In Fig. 25 è rappresentato un sistema a 3 gdl. Noti i valori delle masse e delle rigidezze, calcolare:

1. le 3 pulsazioni naturali del sistema (esprimerle in rad/s);
2. le 3 forme modali (eseguire la normalizzazione in modo che la prima componente sia unitaria).
3. introdotte nel sistema le modifiche strutturali indicate nei dati, calcolare il nuovo valore della seconda pulsazione propria del sistema impiegando il *quoziente di Rayleigh*.

DATI

u ultima cifra del numero di matricola
(matricola $= 0000\#\#\#\#u$)
$m_1 = 2 \times m$
$m_2 = 3 \times m$
$m_3 = 2 \times m$
$k_1 = 4 \times k$
$k_2 = 3 \times k$
$k_3 = 5 \times k$
$m = 1 + u/10$ $[kg]$
$k = 1 - u/10$ $[N/m]$

Modifiche strutturali:
$\Delta m_3 = 0.4 \times m$
$\Delta k_2 = 0.7 \times k$



%% EQUAZIONI DI EQUILIBRIO

Indico la derivata seconda con il tag _2punti

```
1) m1*x1_2punti + k1*x1 +k2*x1 - k2*x2 = 0
2) m2*x2_2punti + k2*x2 + k3*x2 -k2*x1 -k3*x3 = 0
3) m3*x3_2punti + k3*x3 - k3*x2 = 0
```

forma matriciale [M]{X_2punti} + [K]{X}={0}

[M]=

m1	0	0
0	m2	0
0	0	m3

[K]=

K1+k2	-k2	0
-k2	K2+k3	-k3
0	-k3	K3

%% PULSAZIONI NATURALI

Uso il comando eig per trovare la matrice con le forme modali e la matrice dinamica diagonale e di conseguenza le pulsazioni naturali

```
[Mmod,D]=eig(K,M);
```

D = matrice dinamica diagonale
Mmod matrice con forme modali : Mmod= [{X}1 {X}2 {X}3]

```
w1=sqrt(D(1,1)) [rad/s]
w2=sqrt(D(2,2)) [rad/s]
w3=sqrt(D(3,3)) [rad/s]
```

%% NORMALIZZAZIONE FORME MODALI

Normalizzo le forme modali in modo che la prima componente sia unitaria
Ricavo i coefficienti p per normalizzare le forme modali
Dividendo la matrice modale per i coefficienti p ottengo la matrice modale normalizzata

```
p = Mmod(1,:)
P = ones(1,length(Mmod)).*p
Mmod_normalizzata = Mmod./P
```

%% INTRODUZIONE MODIFICHE STRUTTURALI

modifico masse e rigidezze

[ΔM]=

0	0	0
0	0	0
0	0	Δm3

[ΔK]=

Δk2	-Δk2	0
-Δk2	ΔK2	0
0	0	0

modificando masse e rigidezze variano anche i modi e le frequenze

{X}j* = {X}j + delta{X}j

wj* = wj +deltawj

```
%% QUOZIENTE DI RAYLEIGH

cerco la seconda frequenza propria dopo le modifiche

wj*^2= ( {X}jᵀ*[K+ΔK]*{X}j ) / ( {X}jᵀ *[M+ΔM]*{X}j )
--> wj*^2= ( {X}jᵀ *[K]*{X}j + {X}jᵀ *[ΔK]*{X}j ) / ( {X}jᵀ *[M]*{X}j + {X}jᵀ *[ΔM]*{X}j )

ricorda : wj^2 = ( {X}jᵀ *[K]*{X}j ) / ({X}jᵀ *[M]*{X}j)

raccolgo wj^2 per semplificare
```

$$wj *^2 = wj^2 * \frac{\dfrac{1+\{X\}j^T*[\Delta K]*\{X\}j}{1+\{X\}j^T*[K]*\{X\}j}}{\dfrac{1+\{X\}j^T*[\Delta M]*\{X\}j}{1+\{X\}j^T*[M]*\{X\}j}}$$

Ultima cifra del numero di matricola	**Prima frequenza propria [*rad/s*]**	**Seconda frequenza propria [*rad/s*]**	**Terza frequenza propria [*rad/s*]**	**Seconda frequenza propria dopo le modifiche [*rad/s*]**
***u*=4**	0.4971	1.6220	2.0525	1.6136

```
%% DATI

u =1; %ultima cifra del numero di matricola

m = 1 + u/10; % [kg]
k = 1 - u/10; % [N/m]
m1 = 2 * m;
m2 = 3 * m;
m3 = 2 * m;
k1 = 4 * k;
k2 = 3 * k;
k3 = 5 * k;

% Modifiche strutturali:

% variazioni dovute alle modifiche strutturali del modello
deltam3 = 0.4 * m;
deltak2 = 0.7 * k;

%% EQUAZIONI DI EQUILIBRIO

% 1) m1*x1_2punti + k1*x1 +k2*x1 - k2*x2 = 0
% 2) m2*x2_2punti + k2*x2 + k3*x2 -k2*x1 -k3*x3 = 0
% 3) m3*x3_2punti + k3*x3 - k3*x2 = 0

% forma matriciale [M]{X_2punti} + [K]{X}={0}
[M]=[m1 0 0; 0 m2 0; 0 0 m3];
[K]=[(k1+k2) -k2 0; -k2 (k2+k3) -k3; 0 -k3 k3];

%% PULSAZIONI NATURALI

[Mmod,D]=eig(K,M);
% D matrice diagonale con pulsazioni al quadrato
% Mmod matrice con forme modali
% Mmod= [{X}1 {X}2 {X}3 ]

w1=sqrt(D(1,1)) %rad/s
w2=sqrt(D(2,2)) %rad/s
w3=sqrt(D(3,3)) %rad/s

%% NORMALIZZAZIONE FORME MODALI

p = Mmod(1,:);
P = ones(1,length(Mmod)).*p;
Mmod_normalizzata = Mmod./P;

prima_forma_modale_normalizzata_trasposta=(Mmod_normalizzata(
:,1)).';
```

```
seconda_forma_modale_normalizzata_trasposta=(Mmod_normalizzat
a(:,2)).';
terza_forma_modale_normalizzata_trasposta=(Mmod_normalizzata(
:,3)).';

%% INTRODUZIONE MODIFICHE STRUTTURALI

% modificando masse e rigidezze variano anche i modi e le
frequenze
% {X}j* = {X}j + delta{X}j
% wj* = wj +deltawj

[deltaK] = [ deltak2 -deltak2 0; -deltak2 deltak2 0; 0 0 0];
[deltaM] = [ 0 0 0; 0 0 0; 0 0 deltam3 ];

%% QUOZIENTE DI RAYLEIGH
% cerco la seconda frequenza propria dopo le modifiche

% wj*^2= ( {X}j_trasposta*[K+deltaK]*{X}j ) / (
{X}j_trasposta*[M+deltaM]*{X}j )
% --> wj*^2= ( {X}j_trasposta*[K]*{X}j +
{X}j_trasposta*[deltaK]*{X}j ) /
%           / ( {X}j_trasposta*[M]*{X}j +
{X}j_trasposta*[deltaM]*{X}j )

% ricorda : wj^2 = ( {X}j_trasposta*[K]*{X}j ) /
({X}j_trasposta*[M]*{X}j)
% raccolgo wj^2 per semplificare
% wj^2 = wj^2 * (Num/Den)
% Num = 1 + ( {X}j_trasposta*[deltaK]*{X}j ) / (
{X}j_trasposta*[K]*{X}j )
% Den = 1 + ( {X}j_trasposta*[deltaM]*{X}j ) / (
{X}j_trasposta*[M]*{X}j )

seconda_forma_modale_trasposta=(Mmod(:,2)).';
seconda_forma_modale=Mmod(:,2);

Num = 1 + ( seconda_forma_modale_trasposta * deltaK *
seconda_forma_modale ) / ( seconda_forma_modale_trasposta * K
* seconda_forma_modale );
Den = 1 + ( seconda_forma_modale_trasposta * deltaM *
seconda_forma_modale ) / ( seconda_forma_modale_trasposta * M
* seconda_forma_modale );

w2_asterisco =sqrt( w2^2 * (Num/Den) )
```

14 Definizione dei parametri di acquisizione

Si vogliono effettuare rilievi sperimentali di vibrazione sulla struttura rappresentata schematicamente in Fig. 26. La misura va condotta all'interno del campo di frequenze $0 \div f^\star$ e, ai fini dell'analisi, occorre ottenere una risoluzione spettrale massima pari a Δf.

Determinare:

1. La frequenza di taglio del filtro passa basso anti-aliasing
2. La frequenza di campionamento minima
3. Il numero di punti da elaborare tenendo conto che si desidera utilizzare l'algoritmo FFT (Fast Fourier Transform)

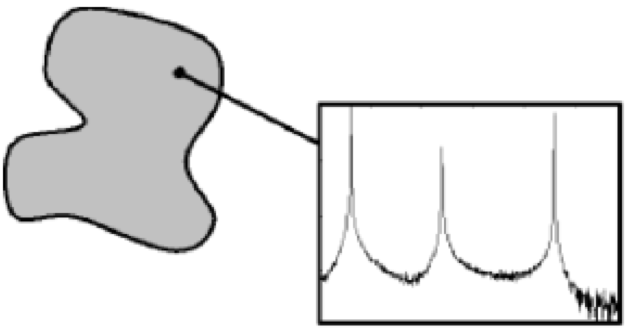


Figura 26: *Misura di vibrazioni*

DATI

u ultima cifra del numero di matricola (matricola $= 0000\#\#\#\#u$)

$f^\star = 4000 - 200 \times u \; [Hz]$
$\Delta f = 10 + 0.5 \times u^2 \; [Hz]$

```
%% FREQUENZA DI TAGLIO FILTRO PASSA BASSO

La misura va condotta all'interno del campo del campo di
frequenze [0;f*] Hz.
Seleziono un filtro passa basso con frequenza di taglio
Ft= f* [Hz]

%% FREQUENZA DI CAMPIONAMENTO MINIMA

La frequenza di campionamento minima deve essere almeno 2
volte la frequenza massima (teorema di Shannon) per stare sul
sicuro la prendo 2.5 volte superiore

Fs_minima = 2.5 * Ft % [Hz]
```

%% NUMERO DI PUNTI DA ELABORARE PER FFT

Ricordo che l'algoritmo FFT elabora solo potenze di 2.

Calcolo il numero di campioni necessari e scelgo come N una potenza di 2 uguale o superiore al numero trovato

Tempo di osservazione: T*=1/deltaf [s]
intervallo di campionamento : deltat=1/Fs_minima [s]
numero di campioni necessari N = T*/deltat

Scelgo una potenza di 2 uguale o superiore a N

Ultima cifra del numero di matricola	**Frequenza di taglio del filtro passa-basso (anti-aliasing) [*Hz*]**		**Frequenza di campionamento minima [*Hz*]**	**Numero di punti da elaborare (tenendo conto che si desidera utilizzare l'algoritmo FFT)**
***u*=4**	3200		8000	512

```
%% DATI
u=4; %ultimo numero matricola
f_asterisco = 4000 - 200 * u; %[Hz] estremo superiore del
campo delle frequenze
deltaf = 10 + 0.5 * u^2; %[Hz] massima risoluzione spettrale

%% FREQUENZA DI TAGLIO FILTRO PASSA BASSO

% La misura va condotta all'interno del campo del campo di
frequenze
% [0;f*] Hz. Selezione un filtro passa basso con frequenza di
taglio Ft= f*

Ft = f_asterisco % frequenza di taglio filtro passa basso
[Hz]

%% FREQUENZA DI CAMPIONAMENTO MINIMA

% la frequenza di campionamento minima deve essere almeno 2
volte la
% frequenza massima (teorema di Shannon) per stare sul sicuro
la prendo 2.5
% volte superiore

Fs_minima = 2.5 * Ft % [Hz]

%% NUMERO DI PUNTI DA ELABORARE PER FFT

T_asterisco=1/deltaf; %tempo di osservazione
deltat=1/Fs_minima; %intervallo di campionamento
N_perfetto= T_asterisco/deltat; %numero di campioni necessari

% Ricordo che l'algoritmo FFT elabora solo potenze di 2

vettore_potenze_2 = [1 2 4 8 16 32 64 128 256 512 1024 2048
4096 8192 16384 32768 65536 131072 262144 524288 ];
indice=find(vettore_potenze_2 >= N_perfetto);
N= vettore_potenze_2(min(indice)) %numero di campioni minimo
per usare FFT
```

TABELLA di RIEPILOGO dei RISULTATI

MECCANISMO PER MACCHINA CUCITRICE

Ultima cifra del numero di matricola	Lx [mm]	Ly [mm]
u= 4	32.1221	27.0586

SINTESI CINEMATICA DI UN QUADRILATERO ARTICOLATO

Resto 2	Coordinate punto A0 [mm]	Coordinate punto B0 [mm]	Lunghezza asta A0A [mm]	Lunghezza biella AB [mm]	Lunghezza asta B0B [mm]	Angolo iniziale manovella [deg]
0	165,8594 - 303,3744 i	462,4280 - 354,6821 i	103,0640	202,8438	307,6107	62.2100

PROPORZIONAMENTO DI UN MECCANISMO A CAMMA

Resto1	Raggio base [mm]	Diametro bicchierino [mm]	Precarico [N]
0	18.7038	31.4839	482,2200

RIDUTTORE A DUE STADI

Resto1	Angolo di pressione di lavoro α3,4 [deg]	Somma coefficienti di spostamento x3 + x4	Somma spostamenti di profilo v3 + v4 [mm]
0	18.1154	-0.9555	-2.8666

FREQUENZA DI UNA COLONNA CON SERBATOIO ELEVATO

Ultime due cifre del numero di matricola		Prima frequenza propria [Hz]
u=8	v=4	0.1539

VIBRAZIONI TORSIONALI DI UN MOTORE MARINO

Ultima cifra del numero di matricola	Prima frequenza propria [Hz]	Seconda frequenza propria [Hz]	Rapporto r1 = [Φ2/Θ1]1	Rapporto r2 = [Φ2/Θ1]2
u=4	1.4666	16.7078	2.1042	-0.3683

MODIFICHE STRUTTURALI

Ultima cifra del numero di matricola	Prima frequenza propria [rad/s]	Seconda frequenza propria [rad/s]	Terza frequenza propria [rad/s]	Seconda frequenza propria dopo le modifiche [rad/s]
u=4	0.4971	1.6220	2.0525	1.6136

DEFINIZIONE DEI PARAMETRI DI ACQUISIZIONE

Ultima cifra del numero di matricola	Frequenza di taglio del filtro passa-basso (anti-aliasing) [Hz]	Frequenza di campionamento minima [Hz]	Numero di punti da elaborare (tenendo conto che si desidera utilizzare l'algoritmo FFT)
u=4	3200	8000	512