



**POLITECHNIKA
RZESZOWSKA**
im. IGNACEGO ŁUKASIEWICZA



**WYDZIAŁ
BUDOWY MASZYN
I LOTNICTWA**
POLITECHNIKI RZESZOWSKIEJ

Geometria i kinematyka ząbów

Laboratorium

Ćwiczenie 2: Powierzchnie

Opracował: dr inż. Michał Batsch

1 Wprowadzenie

W ramach ćwiczenia, na podstawie wiadomości z wykładu należy wygenerować powierzchnię ewolwentowo-śrubową oraz kierunki normalne w wybranych punktach. W tym celu w nowotworzonym pliku w pierwszej kolejności należy zdefiniować parametry koła zębatego (Rys. 1).

```
b=30; % szerokość wieńca zębatego [mm]
mn=5; % moduł [mm]
alfan=20*pi/180; % normalny kąt przyporu [rad]
beta=30*pi/180; % kąt pochylenia linii zębów [rad]

alfat=atan(tan(alfan)/cos(beta)); % czołowy kąt przyporu [rad]
gammat=tan(alfat)-alfat; % involuta kąta przyporu [rad]

mt=mn/cos(beta); % moduł czołowy [mm]
z=19; % liczba zębów [-]
d=mt*z; % średnica podziałowa [mm]
r=d/2; % promień podziałowy [mm]
H=2*pi*r*cot(beta); % skok linii śrubowej [mm]

db=d*cosd(alfat); % średnica zasadnicza [mm]
rb=db/2; % promień zasadniczy [mm]

ha=mn; % wysokość głowy zęba [mm]
hf=1.25*mn; % wysokość stopy zęba [mm]

da=d+2*ha; % średnica wierzchołków [mm]
df=d-2*hf; % średnica dna wębów [mm]

alfata=acos(db/da); % kąt przyporu na średnicy wierzchołków [rad]
gammata=tan(alfata)-alfata; % involuta kąta przyporu na średnicy wierzchołków [rad]

tp=0; % początkowy kąt odtaczania [rad]
tk=gammata+alfata; % końcowy kąt odtaczania [rad]
dt=(tk-tp)/20; % przyrost kąta odtaczania [rad]

fip=0; % początkowy kąt linii śrubowej [rad]
fik=b/(r*cot(beta)); % końcowy kąt linii śrubowej [rad]
dfi=(fik-fip)/30; % przyrost parametru linii śrubowej [rad]
```

Rys. 1 Definicja parametrów koła zębatego

Podobnie jak w poprzednim ćwiczeniu zdefiniowano początkowe i końcowe parametry zarysu jak również parametry linii śrubowej. Następnie generowana jest siatka współrzędnych krzywoliniowych (funkcja „meshgrid”) oraz zgodnie ze wzorami z wykładu definiowany jest wektor powierzchni oraz wektor normalny (Rys. 2).

```

[t,fi]=meshgrid(tp:dt:tk,fip:dfi:fik);

x=rb*(sin(fi + t) - t.*cos(fi + t));
y=rb*(cos(fi + t) + t.*sin(fi + t));
z=fi*r*cot(beta);

nx=1/sqrt(r^2*cot(beta)^2+rb^2)*r*cot(beta)*cos(fi+t);
ny=1/sqrt(r^2*cot(beta)^2+rb^2)*(-r*cot(beta)*sin(fi+t));
nz=1/sqrt(r^2*cot(beta)^2+rb^2)*(-rb)*ones(size(t));

```

Rys. 2 Wektor współrzędnych i współrzędne wersora normalnego

Proszę zwrócić uwagę, że składowa „z” wersora nie zależy od żadnego z parametrów powierzchni. Aby zachować te same wymiary wektorów nx ny nz, wektor nz został przemnożony przez macierz zawierającą same jedyńki o wymiarze równym wymiarowi siatki współrzędnych krzywoliniowych (funkcja „ones” i „size”). Następnie można zwizualizować otrzymane wyniki w postaci powierzchni (funkcja „surf”) oraz wersorów normalnych (funkcja „quiver3”) (Rys. 3)

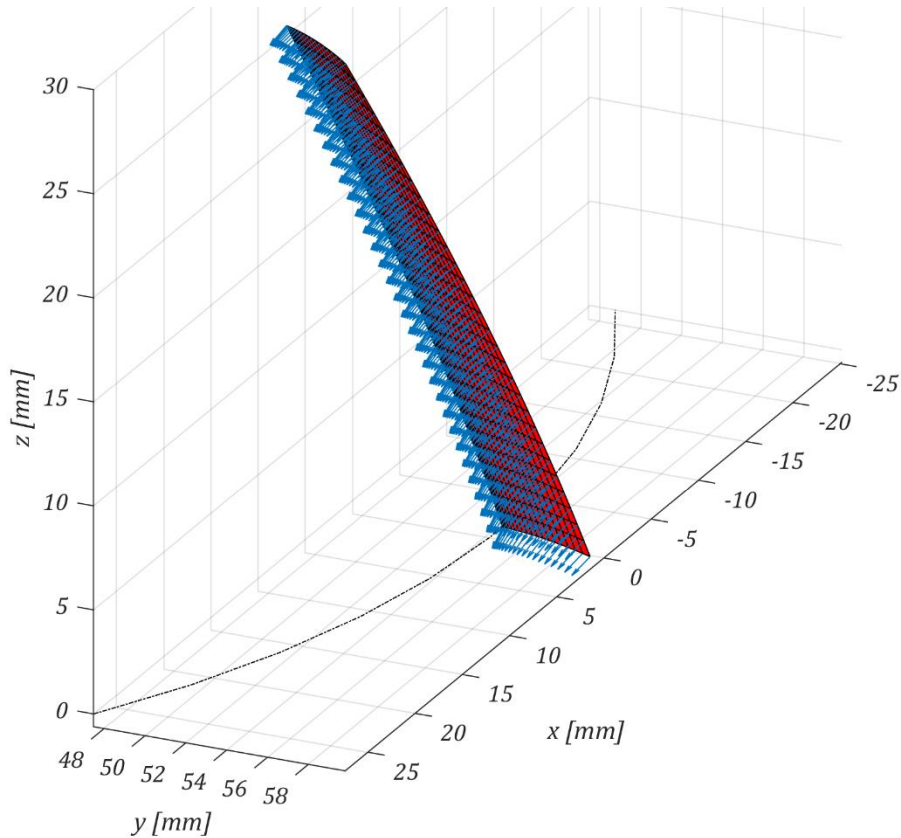
```

figure
surf(x,y,z,'Facecolor','red')
hold on
quiver3(x,y,z,nx,ny,nz)
hold on
f=pi/2-pi/6:0.1:pi/2+pi/6;
plot(rb*cos(f),rb*sin(f),'-.black')
grid on;
xlabel 'x [mm]'
ylabel 'y [mm]'
zlabel 'z [mm]'
grid on
axis equal

```

Rys. 3 Wizualizacja wyników

Rezultat powinien być zbliżony do tego z rysunku 4.



Rys. 4 Wizualizacja wyników

W celu wygenerowania wektorów normalnych nie trzeba znać analitycznego przedstawienia wektora normalnego. Można posłużyć się gotową funkcją „surfnorm” (Rys. 5)

```
[nxx,nyy,nzz]=surfnorm(x,y,z);

figure
surf(x,y,z,'Facecolor','red')
hold on
quiver3(x,y,z,nxx,nyy,nzz)
hold on
plot(rb*cos(f),rb*sin(f),'-black')
grid on;
xlabel 'x [mm]'
ylabel 'y [mm]'
zlabel 'z [mm]'
grid on
axis equal
```

Rys. 5 Funkcja „surfnorm”

Uzyskany rezultat powinien być zbliżony do tego z rysunku 4. Różnice mogą pojawić się na fi-linii dla $t=0$, gdyż funkcja „surfnorm” bazuje na algorytmach numerycznych. Ten problem można wyeliminować poprzez ustawienie wartości $tp=0.1$. Wówczas różnice pomiędzy wygenerowanymi dwoma metodami wektorami normalnymi będą niewielkie.

2 Zadanie

W ramach ćwiczenia, należy samodzielnie napisać program, który będzie generował i pokazywał wersory normalne do powierzchni ewolwentowo-śrubowej na zdefiniowanej przez użytkownika **średnicy i wysokości**.