RotorFF

1 Syntax

[Vt,wt,Vp,wp,Pt,Tp] = RotorFF(alfa)

2 Function description

[Vt,wt,Vp,wp,Pt,Tp] = RotorFF(alfa) returns the characteristic curves for rotor in forward flight for both constant thrust and power and gives in output also the relative x and y values. It requires in input the angle of attack in degrees.

The plot avaiable are:

- For constant Thrust:
  - -w versus V
  - -P versus V
- For constant Power:
  - -w versus V
  - -T versus V

where w = induction, V = asymptotic velocity, T = Thrust, P = Power

3 Algorithm description

The algorithm implements the following equations:

Constant Thrust

- $\left(\tilde{V}_\infty^2 \tilde{w} \sin\alpha + \tilde{w}^2\right)^2 + \tilde{V}_\infty^2 \tilde{w}^2 \cos^2\alpha = 1$
- $\tilde{P}_i = \widetilde{V_\infty} \sin\alpha + \tilde{w}$

Constant Power

- $\left[\left(\tilde{V}_\infty^2 \tilde{w} \sin\alpha + \tilde{w}^2\right)^2 + \tilde{V}_\infty^2 \tilde{w}^2 \cos^2\alpha\right]\left(\tilde{V}_\infty \sin\alpha + \tilde{w}\right) = 1$
- $\tilde{T} = \left(\tilde{V}_\infty \sin\alpha + \tilde{w}\right)^{-1}$

The code begins with the function call.

```
function [Vt,wt,Vp,wp,Pt,Tp] = RotorFF(alfa)
```

The angle of attack given as input in degrees is converted in radiant.

```
alfa=deg2rad(alfa);
```

The two implicit functions are managed through a function handle.

```
f1 = @(x,y) (((x.*cos(alfa)).^2 + (x.*sin(alfa)+y).^2).*(y.^2))-1;
f2 = @(x,y) (((x.*cos(alfa)).^2 +…
(x.*sin(alfa)+y).^2)).*(y.^2).*((x.*sin(alfa)+y))-1 ;
```

They are plotted using the **fimplicit** function.

```
h1=fimplicit(f1,[0 10 0 1],'k');
h2=fimplicit(f2,[0 5 0 1],'k');
```

After that, the values of the axis are extracted through the **get** function.

```
Vt = (get(h1, 'XData'));
wt = (get(h1, 'YData'));

Vp= (get(h2, 'XData'));
wp= (get(h2, 'YData'));
```

These values are used to define the remaining function and they are also available in output.

```
Pt=Vt.*sin(alfa)+wt;
Tp=(Vp*sin(alfa)+wp).^(-1);
```

These two functions are finally plotted.

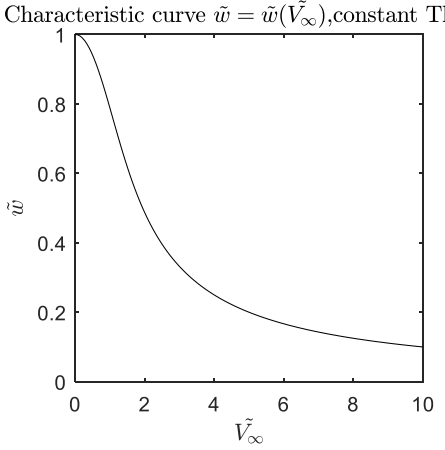
```
plot(Vt,Pt,'k')
plot(Vt,Pt,'k')
```

4 Examples

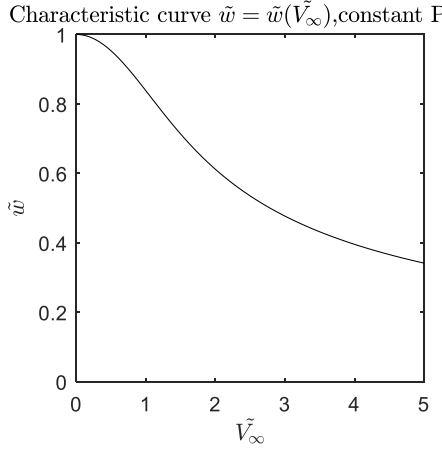
```
alfa=0;

[Vt,wt,Vp,wp,Pt,Tp] = RotorFF(alfa)
```

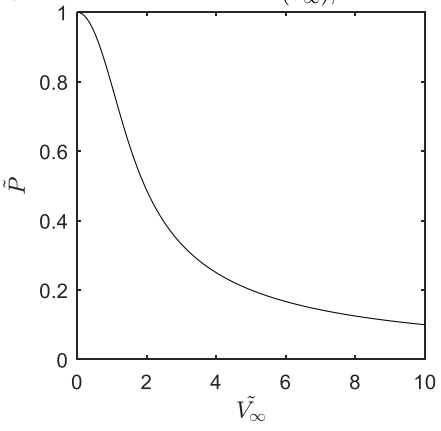
Characteristic curve $\tilde{w} = \tilde{w}(\tilde{V_\infty})$,constant Thrust



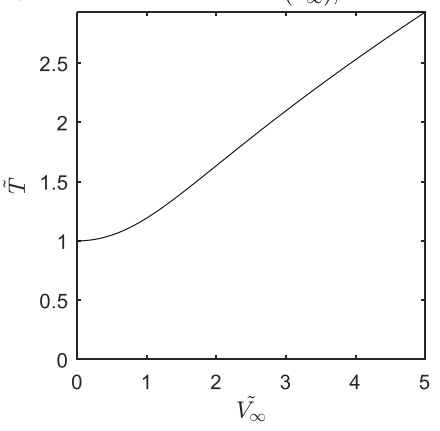
Characteristic curve $\tilde{w} = \tilde{w}(\tilde{V_\infty})$,constant Power



Characteristic curve $\tilde{P} = \tilde{P}(\tilde{V_\infty})$,constant Thrust



Characteristic curve $\tilde{T} = \tilde{T}(\tilde{V_\infty})$,constant Power



5 Input Arguments

Alfa – angle of attack

6 Output Arguments

Constant Thrust

Vt – asymptotic velocity
wt – induction
Pt– power

Constant Power

Vp – asymptotic velocity
wp– induction
Tt– power

All the values are non-dimensional in respect to their value in hovering (for V is used induction in hovering).

7 Reference

R. Tognaccini. "Lezioni di aerodinamica dell'ala rotante" *2019* pp. 84-85.

8 Complete Script

```
function [Vt,wt,Vp,wp,Pt,Tp] = RotorFF(alfa)

%%Angle conversion to radiant
alfa=deg2rad(alfa);

%% Constant thrust
%Function definition
f1 = @(x,y) (((x.*cos(alfa)).^2 + (x.*sin(alfa)+y).^2).*(y.^2))-1;

%Plot setting
subplot(2,2,1);
h1=fimplicit(f1,[0 10 0 1],'k');
axis square;
xlabel('$\tilde{V_{\infty}}$','Interpreter','latex');ylabel('$\tilde{w}$','Interp
reter','latex');
title('Characteristic curve $\tilde{w} = \tilde{w}(\tilde{V_{\infty}})$,constant
Thrust', 'Interpreter','latex') ;

%Axis values
Vt = (get(h1, 'XData'));
wt = (get(h1, 'YData'));

%Function of Power versus speed for costant Thrust
Pt=Vt.*sin(alfa)+wt;
```

```
%Plot setting
subplot(2,2,3);
plot(Vt,Pt,'k')
axis([0 max(Vt) 0 max(Pt)]);
axis square;
xlabel('$\tilde{V_{\infty}}$','Interpreter','latex');ylabel('$\tilde{P}$','Interp
reter','latex');
title('Characteristic curve $\tilde{P} = \tilde{P}(\tilde{V_{\infty}})$,constant
Thrust', 'Interpreter','latex') ;

 %% Constant Power

%Function Definition
f2 = @(x,y) (((x.*cos(alfa)).^2 +…
(x.*sin(alfa)+y).^2)).*(y.^2).*((x.*sin(alfa)+y))-1 ;

%Plot setting
subplot(2,2,2);
h2=fimplicit(f2,[0 5 0 1],'k');
axis square;
xlabel('$\tilde{V_{\infty}}$','Interpreter','latex');ylabel('$\tilde{w}$','Interp
reter','latex');
title('Characteristic curve $\tilde{w} = \tilde{w}(\tilde{V_{\infty}})$,constant
Power', 'Interpreter','latex') ;

%Axis values
Vp= (get(h2, 'XData'));
wp= (get(h2, 'YData'));

%Function of Thrust versus speed for costant Power
Tp=(Vp*sin(alfa)+wp).^(-1);

%Plot setting
subplot(2,2,4);
plot(Vt,Pt,'k')
axis([0 max(Vp) 0 max(Tp)]);
axis square;
xlabel('$\tilde{V_{\infty}}$','Interpreter','latex');ylabel('$\tilde{T}$','Interp
reter','latex');
title('Characteristic curve $\tilde{T} = \tilde{T}(\tilde{V_{\infty}})$,constant
Power', 'Interpreter','latex') ;

end
```