Eduard Algar

Rough Estimation of the Apache AH-64 Main Rotor Friction Torque and Power

1. Calculate blade tip (and root) speed ut (ur), Mach number Mt (Mr) and chord-based Reynolds
number Ret (Rer). Discuss the validity of the assumptions on incompressibility.

We will consider the blades as a solid rigid rotating with the same ω, therefore:

$$u = \omega \cdot r$$

Obtaining the velocity u, it is possible to compute the Mach number:

$$M = \frac{u}{\sqrt{\gamma \cdot R \cdot T}}$$

```
from sympy import *;import matplotlib.pyplot as pyplot; import numpy as np

Rr=2.032
Rt=7.3152
rpm=292
omega = 292*(2*pi/60)
rho=1.225
gamma =1.4
Rair = 287.14
T=288.15
nu= 1.4531e-5
a = sqrt(gamma*Rair *T)
L=0.5334
r= np.arange(0,Rt+0.1,0.1)
v=r*omega
pyplot.plot(r,v)
pyplot.xlabel('radius [m]')
pyplot.ylabel('V[m/s]')
pyplot.scatter(Rr,0,marker='$\u2193$',s=100)
pyplot.text(Rr,1.5,r'$R\! _{r}$')
pyplot.title(r'$u\! _{t}$')
pyplot.xticks(np.arange(0, Rt, 0.5))
pyplot.show()
print('V_= ',float(v[-1]),'m/s')
```

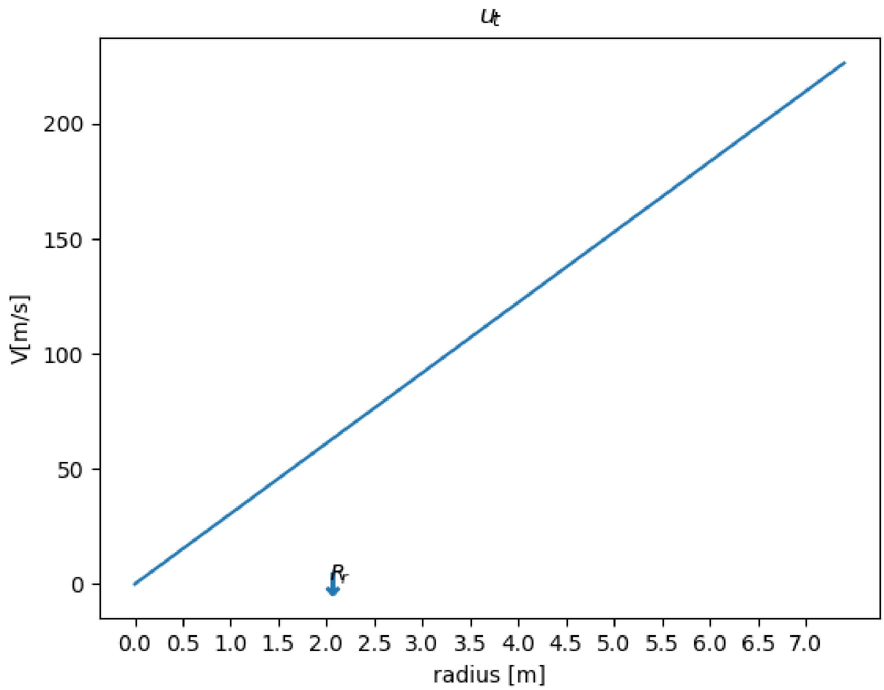

fig 1 Linear velocity distribution

$$u_t = 226.27845 \text{ m/s}$$

```
M = v/a
M03= np.ones_like(M)*0.3
pyplot.plot(r,M, label='M')
pyplot.plot(r,M03,'--',label='compressible regime trasition', color='r')
pyplot.xlabel('radius [m]')
pyplot.ylabel('M')
pyplot.plot(Rr,0,marker='$\u2193$')
pyplot.text(Rr,0.01,r'$R\! _{r}$')
pyplot.title('Mach')
pyplot.legend()
pyplot.xticks(np.arange(0, Rt, 0.5))
pyplot.show()
```

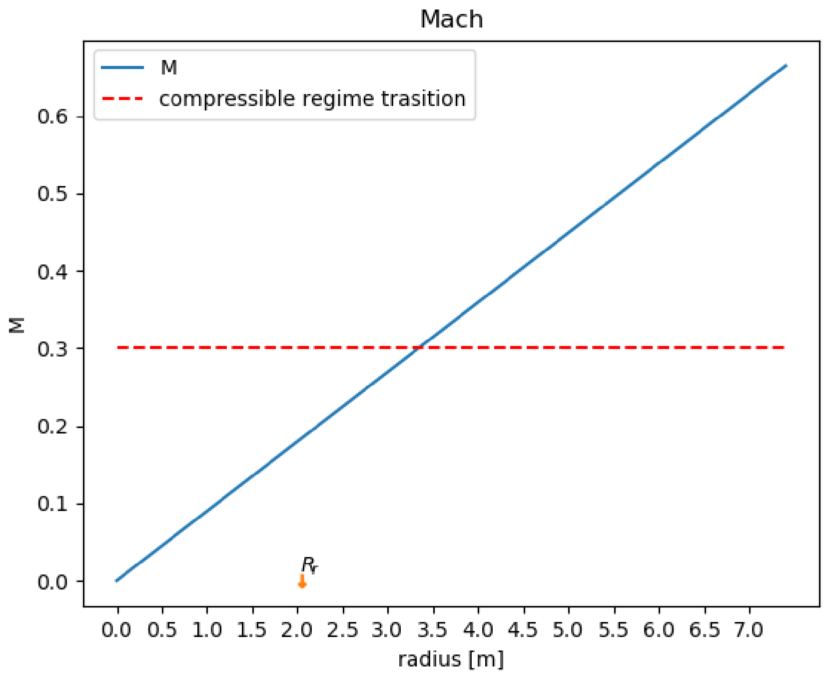

fig 2Mach distribution

M = 0.664849

From the previous plot, it is shown clearly that incompressibility assumptions are not valid as most of the blade is experiencing M>0.3

2. Assuming a natural turbulent transition Reynolds number ReT = $5 \cdot 10^5$***, assess at which radius RT transition starts occurring naturally along the blade-section chord.***

From Reynolds number definition:

$$\mathrm{Re_L} = \frac{\mathrm{u} \cdot \mathrm{L}}{\nu}$$

```
ReL = []
for i in range(len(v)):
    reynolds= (L*v[i])/nu
    ReL.append(reynolds)
Ret= np.ones_like(v)*5e5
pyplot.plot(r,ReL,label=r'$R\! _{L}$',color='b')
pyplot.plot(r,Ret,'--',color='r',label='Turbulent trasition: '+r'$R\!
_{t}=5\cdot 10^{5}$')
pyplot.xlabel('radius [m]')
pyplot.ylabel(r'$Re$')
pyplot.scatter(Rr,6.5e5,marker='$\u2193$',s=100,color='green')
pyplot.text(Rr+0.1,6.5e5,r'$R\! _{r}$')
pyplot.title(r'$Re$')
pyplot.xticks(np.arange(0, Rt, 0.5))
pyplot.grid()
pyplot.legend()
pyplot.show()
```

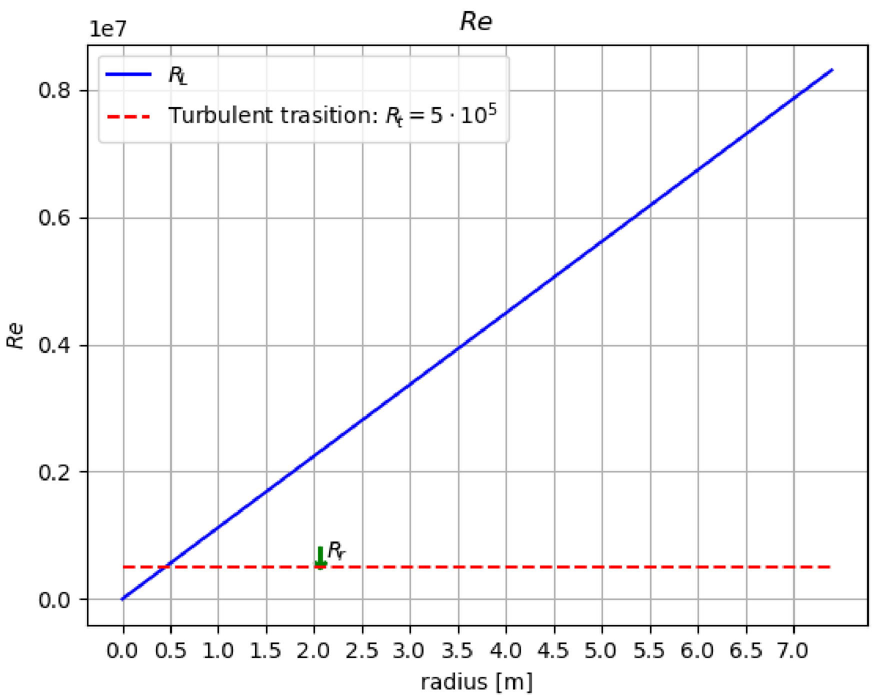

fig 3Evolution of Reynolds number along the radius

From the graph the turbulent flow starts to take place at $r{\sim}0.445m$

3. Express the boundary layer trailing edge momentum thickness (θte=L)(r) in terms of tip radius Rt, tip Reynolds number Ret and distance r to blade hub.

Thanks to the flat plate hypothesis, we are now in position to exploit the results for the Integral method (FP):

$$p = \frac{k_1}{Re_x^{s_0}} \qquad p = \frac{k_2}{Re_\theta^{m_0}}$$

	Laminar $m_0 = 1$ $s_0 = 1/2$		Turbulent $m_0 = 1/5$ $s_0 = 1/6$	
p	k_1	k_2	k_1	k_2
θ/x	0.664	0.441	0.0221	0.0103
δ_1/x	1.721	1.143	0.0309	0.0144
C_f	0.664	0.441	0.0368	0.0172

Therefore, if x =L:

$$\frac{\theta_{te}}{L} = \frac{0.0221}{\mathrm{Re_L}^{1/6}} \rightarrow \frac{0.0221}{\left(\frac{\omega \cdot \mathrm{R}}{\nu}\right)^{1/6} \left(\frac{\mathrm{R_T}}{\mathrm{R_T}}\right)^{1/6}} \longrightarrow \frac{0.0221}{\left(\frac{\omega \cdot \mathrm{R_T}}{\nu}\right)^{1/6} \cdot \left(\frac{R}{\mathrm{R_T}}\right)^{1/6}}$$

$$\boxed{\frac{\theta_{te}}{L} = \frac{0.0221}{\mathrm{Re_T}^{1/6}} \cdot \left(\frac{R}{\mathrm{R_T}}\right)^{-1/6}}$$

```
tetha_L = []

for i in ReL:
    if float(i)==0:
        i = 1e-6
    tetha_L.append(0.0221/(float(i))**(1/6))

pyplot.ylim(0.0015,0.0035)
pyplot.xlim(Rr, Rt)
pyplot.xticks(np.arange(Rr,Rt+0.5,0.55))
pyplot.plot(r,tetha_L,color='b')
pyplot.ylabel(r'$\frac{\theta _{te}}{L}$',fontsize=20)
pyplot.xlabel('radius [m]')
pyplot.scatter(Rr,0.00196,marker='$\u2193$',s=100,color='r')
pyplot.text(Rr+0.1,0.00196,r'$R\! _{r}$')
pyplot.show()
```

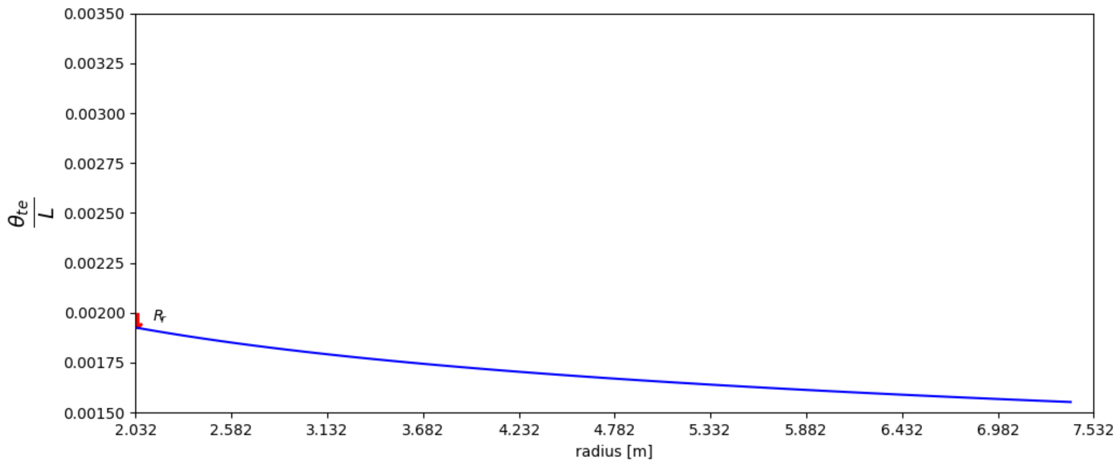

fig 4 Momentum thickness evolution along the blade

4. Calculate the drag per unit span D(r) along the blade. Express it in terms of Rt, Ret, L and Ω.

By definition:

$$D = \int_0^L \tau_w \, dx \xrightarrow{\tau_w = 0.5 \cdot \rho_e \cdot u^2{}_e \cdot Cf} D = \int_0^L \frac{1}{2} \cdot \rho_e \cdot u^2{}_e \cdot Cf \, dx$$

Using the integral momentum eq.:

$$\frac{d\theta}{dx} + \theta \left(\frac{H+2}{u_e} \frac{du_e}{dx} \right) = \frac{C_f}{2}$$

FP

And recasting the previous result:

$$D = \rho_e \cdot (R \cdot \omega)^2 \cdot \int_{\theta_{(0)}=0}^{\theta_{(L)}=\theta_{te}} d\theta$$

$$D = \rho_e \cdot (R \cdot \omega)^2 \cdot \frac{0.0221}{Re_T{}^{1/6}} \cdot \left(\frac{R}{R_T} \right)^{-1/6}$$

```
r = Symbol('r')
Ret = (omega*r*Rt)/nu
D=rho*(omega*r)**2*(0.0221*L/(Ret*r/Rt)**(1/6))
rr=np.arange(Rr,Rt+0.01,0.01)
D_plot=[]
for i in rr:
    D_plot.append(D.subs(r,i))
pyplot.xticks(np.arange(0, Rt, 0.5))
pyplot.xlabel('radius [m]')
pyplot.ylabel('D [N]')
pyplot.plot(rr,D_plot)
pyplot.scatter(Rr,0.00196,marker='$\u2193$',s=100,color='r')
pyplot.text(Rr+0.1,0.00196,r'$R\! _{r}$')
pyplot.title('Drag (one sided plate)')
pyplot.show()
```

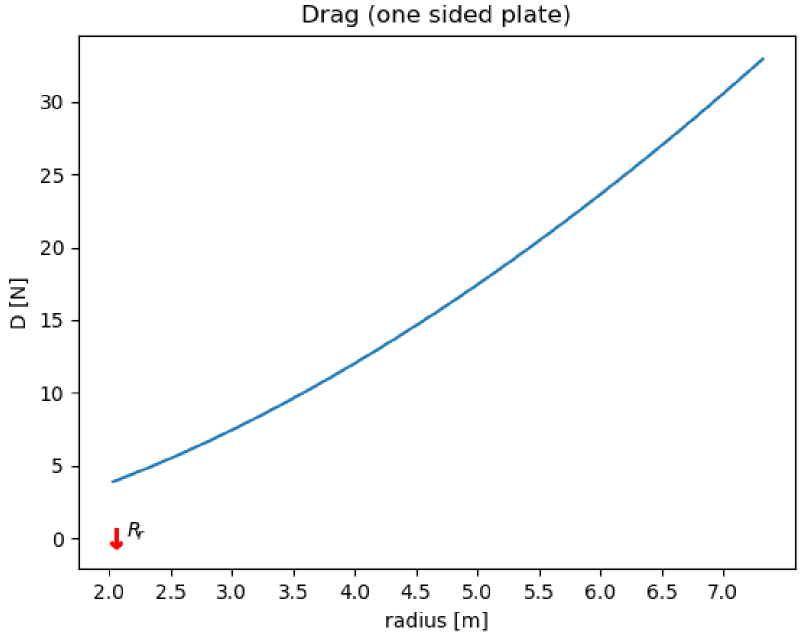

fig 5 Drag distribution for a Flat Plate (one sided)

5. Find the friction torque and power required to spin the rotor in the conditions being considered.

Finally, using the previous results:

$$\tau = \int_0^L \mathrm{D} \cdot \mathrm{r}\,\mathrm{dr}$$

$$P = \tau \cdot \omega$$

```
Df = 4*2*D
Rr=2.032
Rt=7.3152
torque = integrate(Df*r,r)
torque= float(torque.subs(r,Rt)-torque.subs(r,Rr))
print('Torque: ',torque,'N·m')
P = float(omega*torque)
print('Power: ',P,'W')
```

$$\tau = 3803.807894\ \mathrm{N} \cdot \mathrm{m}$$

$$P = 116313.47873\ \mathrm{W}$$