Scilab Textbook Companion for Fluid Mechanics And Thermodynamics Of Turbo Machinery by S. L. Dixon[1]

Created by
Jagadeeshwar Goshika
B.Tech.
Others
IIT Bombay
College Teacher
None
Cross-Checked by
Bhavani Jalkrish

June 9, 2016

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Textbook Companion and Scilab codes written in it can be downloaded from the "Textbook Companion Project" section at the website http://scilab.in

Book Description

Title: Fluid Mechanics And Thermodynamics Of Turbo Machinery

Author: S. L. Dixon

Publisher: Butterworth-Heinemann

Edition: 5

Year: 2005

ISBN: 0-7506-7870-4

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes

Chapter 2

Basic Thermodynamics and Fluid Mechanics

Scilab code Exa 2.1 Ex 1

```
clear all;
clc;
funcprot(0);

//given data
gamma = 1.4;
pi = 8;//pressure ratio
T01 = 300;//inlet temperature in K
T02 = 586.4;//outlet temperature in K

//Calculations
//Calculation of Overall Total to Total efficiency
Tot_eff = ((pi^((gamma-1)/gamma))-1)/((T02/T01)-1);

//Calculation of polytropic efficiency
Poly_eff = ((gamma-1)/gamma)*((log(pi))/log(T02/T01)
    );

//Results
```

```
printf('The Overall total-to-total efficiency is %.2f.\n',Tot_eff);
printf('The polytropic efficiency is %.4f.',Poly_eff);
```

Scilab code Exa 2.2 Ex 2

```
clear all;
clc;
funcprot(0);

//given data
T01 = 1200;//Stagnation temperature at which gas enters in K
p01 = 4;//Stagnation pressure at which gas enters in bar
c2 = 572;//exit velocity in m/s
p2 = 2.36;//exit pressure in bar
Cp = 1.160*1000;//in J/kgK
gamma = 1.33

//calculations
T2 = T01 - 0.5*(c2^2)/Cp;//Calculation of exit temperature in K
Noz_eff = ((1-(T2/T01))/(1-(p2/p01)^((gamma-1)/gamma)));//Nozzle efficiency

//Results
printf('Nozzle efficiency is %.4f.',Noz_eff);
```

Scilab code Exa 2.3 Ex 3

```
clear all;
```

```
clc;
funcprot(0);

//given data
cp = 0.6;//coefficient of pressure
AR = 2.13;//Area ratio
N_R1 = 4.66;

//calculations
cpi = 1 - (1/(AR^2));
Diff_eff = cp/cpi;//diffuser efficiency
theta = 2*(180/%pi)*atan((AR^0.5 - 1)/(N_R1));//included cone angle

//Results
printf('cpi = %.2f.\n',cpi);
printf('The included cone angle can be found = %.2f deg.',theta);
```

Scilab code Exa 2.4 Ex 4

```
clear all;
clc;
funcprot(0);

//given data
AR = 1.8;//Area ratio
cp = 0.6;//coefficient of pressure
N_R1 = 7.85;

//calculations
Theta = 2*(180/%pi)*atan((AR^0.5 - 1)/(N_R1));//included cone angle
cpi = 1-(1/(AR^2));
Diff_eff = cp/cpi;//diffuser efficeincy
```

```

//Results
printf('The included cone angle can be found = %.1f
   deg.\n',Theta);
printf('cpi = %.2f.\n',cpi);
printf('Diffuser efficiency = %.2f.',Diff_eff);
```

Scilab code Exa 2.5 Ex 5

```
clear all;
clc;
funcprot(0);

//given data
AR = 2.0;//Area ratio
alpha1 = 1.059;
B1 = 0.109;
alpha2 = 1.543;
B2 = 0.364;
cp = 0.577;//coefficient of pressure

//calculations
cp = (alpha1 - (alpha2/(AR^2))) - 0.09;
Diff_eff = cp/(1-(1/(AR^2)));//Diffuser efficiency

//Results
printf('The diffuser efficiency = %.4f.',Diff_eff);
```

Chapter 3

Two dimensional Cascades

Scilab code Exa 3.1 Ex 1

```
clear all;
clc;
funcprot(0);

//function to calculate m and delta
function [m,delta] = func(a_l,alpha2,theta)
    m = 0.23*(2*a_l)^2 + alpha2/500;
    delta = m*theta;
endfunction

//given data
alpha1_ = 50;// in deg
alpha2_ = 20;// in deg
a_l = 0.5;//percentage
s_l = 1.0;
eps = 21;//in deg

//Calculations
theta = alpha1_ - alpha2_;
alpha21 = 20;//in deg
[m1,delta1] = func(a_l,alpha21,theta);
```

```
alpha22 = 28.1;//in deg
[m2,delta2] = func(a_l,alpha22,theta);
alpha23 = 28.6;//in deg
[m3,delta3] = func(a_l,alpha23,theta);
alpha1 = eps + alpha23;
i = alpha1 - alpha1_;
alpham = (180/%pi)*atan(0.5*(tan(alpha1*%pi/180) +
    tan(alpha23*%pi/180)));
CL = 2*(s_l)*cos(alpham*%pi/180)*(tan(alpha1*%pi
    /180) - tan(alpha23*%pi/180));

//Results
printf('The fluid deflection = %d deg.',eps);
printf('\n The fluid deviation = %.1f deg.',i);
printf('\n The ideal lift coefficient at the design
    point = %.2f',CL);
```

Scilab code Exa 3.2 Ex 2

```
clear;
clc;
funcprot(0);

//given data
s_l = 1.0;
alpha1_ = 50;//in deg
alpha2_ = 20;//in deg
eps_ = 21;//in deg
i_ = -0.4;//in deg
i = 3.8;//in deg
CD = 0.017;
eps = 1.15*eps_;

//Calculations
alpha1 = alpha1_+i;
```

```
alpha2 = alpha1-eps;
alpham = (180/%pi)*atan(0.5*(tan(alpha1*%pi/180) + tan(alpha2*%pi/180)));
zeta = CD/((s_l)*(cos(alpham*%pi/180))^3);
Cf = 2*(tan(alpha1*%pi/180) - tan(alpha2*%pi/180));
eff_D = 1 - zeta/(Cf*tan(alpham*%pi/180));

//Results
printf('The tangential lift force coefficient = %.3f',Cf);
printf('\n The diffuser efficiency = %d percentage.',eff_D*100);
```

Scilab code Exa 3.3 Ex 3

```
clear;
clc;
funcprot(0);

//given data
alpha1 = 58;//in deg
alpha2 = 44;//in deg
AVR = 1.0;

//Calculations
alpham = (180/%pi)*atan(0.5*(tan(alpha1*%pi/180) + tan(alpha2*%pi/180)));
zetam = (180/%pi)*atan(tan(alpham*%pi/180) - 0.213);
Cpi = 1-(cos(alpha1*%pi/180)/cos(alpha2*%pi/180))^2;
s_l = 9*(0.567-Cpi);
theta = ((zetam-alpha2+1.1*(s_l)^(1/3))/(0.5-0.31*(s_l)^(1/3)));
delta = alpha2-zetam-0.5*theta;

//Results
```

```
printf('The suitable space chord ratio = %.4f',s_l
   );
printf('\n The suitable blade camber = %.2f deg.',
   theta);
```

Chapter 4

Axial flow Turbines Two dimensional Theory

Scilab code Exa 4.1 Ex 1

```
clear all;
clc;
funcprot(0);

//given data
phi = 0.4;
epsilon = 28.6;//in deg

//calculations
alpha2 = (180/%pi)*atan(1/phi);//in deg
zeta = 0.04*(1+ 1.5*(alpha2/100)^2);
eta = 1 + (phi^2)*(zeta*((1/cos(%pi*alpha2/180))^2)
    +0.5);

//results
printf('The efficiency = %.3f.\n',1/eta);
printf('This value appears to be the same as the
    peak value of efficiency curve.\n');
```

Scilab code Exa 4.2 Ex 2

```
clear all;
clc;
funcprot(0);

//given data
alpha2 = 70;//in deg
p01 = 311;//in kPa
T01 = 850;//in degC
p3 = 100;//in kPa
eff_tot_stat = 0.87;
U = 500;//in m/s
Cp = 1.148;//in kJ/(kgC)
gamma = 1.33;

//Calculations
delW = eff_tot_stat*Cp*(T01+273.15)*(1-(p3/p01)^((gamma-1)/gamma));//specific work
cy2 = delW*1000/U;//in m/s
c2 = cy2/sin(%pi*alpha2/180);//in m/s
T2 = (T01+273.15) - 0.5*(c2^2)/(Cp*1000);//Nozzle exit temperature in K
M2 = c2/sqrt(gamma*287*T2);//Nozzle exit mach number
cx = c2*cos(%pi*alpha2/180);//axial velocity in m/s
eff_tot_tot = 1/((1/eff_tot_stat)-((cx^2)/(2*1000*delW)));//Total to total efficiency
R = 1 - 0.5*(cx/U)*tan(%pi*alpha2/180);//stage reaction

//results
printf('(i) The specific work done = %d kJ/kg.\n',delW);
printf('(ii) The Mach number leaving the nozzle = %
```

```
.2f.\n',M2);
printf('(iii) The axial velocity = %d m/s.\n',cx);
printf('(iv) The total-to-total efficiency = %.2f.\n
    ',eff_tot_tot);
printf('(v) The stage reaction = %.3f.\n',R);


//there are small errors in the answers given in the
    book
```

Scilab code Exa 4.3 Ex 3

```
clear all;
clc;
funcprot(0);

//given data
H_b = 5.0;//average bladeaspect ratio for the stage
t_c = 0.2;//max. blade thickness to chord ratio
Re = 1*10^5;//average Reynolds number
cx = 200;//in m/s
cy2 = 552;//in m/s
U = 500;//in m/s
c2 = 588;//in m/s
delW = 276;//in kJ
c3 = 200;//in m/s
Cp = 1.148;//in kJ/(kgC)
T2 = 973;//in K
T01 = 1123;//in K
alpha1 = 0;//in deg
alpha2 = 70;//in deg

//calculations
eps = alpha1 + alpha2;//in deg
zetaN = 0.04*(1 + 1.5*(eps/100)^2);
```

```
zetaN1 = (1+zetaN)*(0.993 + 0.021/H_b) - 1;
beta2 = (180/%pi)*atan((cy2-U)/cx);
beta3 = (180/%pi)*atan(U/cx);
epsR = beta2 + beta3;
zetaR = 0.04*(1 + 1.5*(epsR/100)^2);
zetaR1 = (1+zetaR)*(0.975 + 0.075/H_b) - 1;
w3_U = sqrt(1+(cx/U)^2);
eff_ts = 1/(1 + (zetaR1*w3_U + zetaN1*((c2/U)^2) + (cx/U)^2)/(2*cy2/U));
T3 = T01 - (delW*1000 + 0.5*c3^2)/(Cp*1000);
eff_ts1 = 1/(1 + (zetaR1*(w3_U)^2 + (T3/T2)*zetaN1*((c2/U)^2) + (cx/U)^2)/(2*cy2/U));

//Results
printf('The total-to static efficiency = %.3f.', eff_ts);
printf('\n The result is very close to the value assumed in first example.')
printf('\n The total-to-static efficiency after including the temperature ratio in the equation = %.3f.',eff_ts1);

//there are small errors in the answers given in the book
```

Scilab code Exa 4.4 Ex 4

```
clear;
clc;
funcprot(0);

//given data
T02 = 1200;//in K
p01 = 4.0;//in bar
dt = 0.75;//tip diameter in m
```

```
hb = 0.12;//blade height in m
v = 10500;//shaft speed in rev/min
R = 0.5;//degree of reaction at mean radius
phi = 0.7;//flow coefficient
psi = 2.5;//stage loading coefficient
eff_noz = 0.96;//Nozzle efficiency
Cp = 1160;//in kJ/(kgC)
gamma = 1.33;
Rg = 287.8;//specific gas constant
A2 = 0.2375;//in m^2
K = 2/3;//stress taper factor
rho = 8000;//in kg/m^3

//calculations
beta3 = (180/%pi)*atan((0.5*psi + R)/phi);
beta2 = (180/%pi)*atan((0.5*psi - R)/phi);
alpha2 = beta3;
alpha3 = beta2;
rm = (dt-hb)/2;
Um = (v/30)*%pi*rm;
cx = phi*Um;
c2 = cx/(cos(alpha2*%pi/180));
T2 = T02 - 0.5*(c2^2)/Cp;
p2 = p01*((1-((1-(T2/T02))/eff_noz))^(gamma/(gamma
    -1)));
mdot = ((p2*10^5)/(Rg*T2))*A2*cx;
Ut = (v/30)*%pi*0.5*dt;
sig_rho = K*0.5*(Ut^2)*(1-((dt-2*hb)/dt)^2);
sig = rho*sig_rho;
Tb = T2 + 0.85*((cx/cos(beta2*%pi/180))^2)/(2*Cp);

//Results
printf('(i)The relative and absolute angles for the
    flow: \n beta3 = %.1f deg, and beta2 = %.2f deg.'
    ,beta3,beta2);
printf('\n alpha2 = %.1f deg, and alpha3 = %.2f deg.
    ',alpha2,alpha3);
printf('\n (ii) The velocity at nozzle exit = %.2f m
```

```
/s',c2);
printf('\n (iii)The static temperature and pressure at nozzle exit assuming a nozzle efficiency of %.2f: \n T2 = %.1f K\n p2 = %.3f bar',eff_noz,T2,p2);
printf('\n and mass flow = %.1f kg/s',mdot);
printf('\n (iv)The rotor blade root stress assuming the blade is tapered with a stress taper factor K of 2/3 and \n the blade material density is %d kg/m2 = %.1f MPa',rho,sig/(10^6));
printf('\n (v) The approximate average mean blade temperature is Tb = %.1f K',Tb);
printf('\n (vi)Inspection of the data for Inconel 713 cast alloy suggests that it might be a better choice \n of blade material as the temperature stress point of the above calculation is to the \n left of the line marked creep strain of 0.2 percentage in 1000 hr.')


//there are very small errors in the answers given in textbook
```

Chapter 5

Axial flow Compressors and Fans

Scilab code Exa 5.1 Ex 1

```
clear;
clc;
funcprot(0);

//given data
T01 = 293;//in K
pi = 5;//pressure ratio
R = 0.5;//stage reaction
Um = 275;//in m/s
phi = 0.5;//flow coefficient
psi = 0.3;//stage loading factor
eff_stage = 0.888;//stage efficiency
Cp = 1005;//J/(kgC)
gamma = 1.4;

//Calculations
beta1 = (180/%pi)*atan((R + 0.5*psi)/phi);
beta2 = (180/%pi)*atan((R - 0.5*psi)/phi);
alpha2 = beta1;
```

```
alpha1 = beta2;
delT0 = psi*(Um^2)/Cp;
N = (T01/delT0)*((pi^((gamma-1)/(eff_stage*gamma))) - 1);
N = ceil(N);
eff_ov = ((pi^((gamma-1)/gamma)) - 1)/((pi^((gamma-1)/(eff_stage*gamma))) - 1);
printf('The flow angles are: beta1 = alpha2 = %.2f deg and beta2 = alpha1 = %d deg.',beta1,ceil(beta2));
printf('\n The number of stages required = %d',N);
printf('\n The overall efficiency = %.1f percentage',eff_ov*100);

//there is a small error in the answer given in textbook
```

Scilab code Exa 5.2 Ex 2

```
clear;
clc;
funcprot(0);

//given data
R = 0.5;//stage reaction
s_c = 0.9;//space-chord ratio
beta1_ = 44.5;//in deg
beta2_ = -0.5;//in deg
h_c = 2.0;//height-chord ratio
lamda = 0.86;//work done factor
i = 0.4;//mean radius relative incidence
rho = 3.5;//density in kg/m^3
Um = 242;//in m/s
eps_max = 37.5;//in deg
eps = 37.5;//in deg
```

```
delp0 = 0.032;//the profile total pressure loss coefficient
//Calculations
theta = beta1_ - beta2_;
deltaN = (0.229*theta*(s_c^0.5))/(1 - (theta*(s_c^0.5)/500));
beta2N = deltaN + beta2_;
eps_ = 0.8*eps_max;
i_ = beta2N + eps_ - beta1_;
i = 0.4*eps_ + i_;
beta1 = beta1_ + i;
beta2 = beta1 - eps;
alpha2 = beta1;
alpha1 = beta2;
phi = 1/(tan(alpha1*%pi/180) + tan(beta1*%pi/180));
psi = lamda*phi*(tan(alpha2*%pi/180) - tan(alpha1*%pi/180));
betam = (180/%pi)*atan(0.5*(tan(beta1*%pi/180) + tan(beta2*%pi/180)));
CL = 2*s_c*cos(betam*%pi/180)*(tan(beta1*%pi/180) - tan(beta2*%pi/180));
CDp = s_c*(delp0)*((cos(betam*%pi/180))^3)/((cos(beta1*%pi/180))^2);
CDa = 0.02*s_c/h_c;
CDx = 0.018*CL^2;
CD = CDp + CDa + CDx;
eff_tt = 1 - (CD*phi^2)/(psi*s_c*((cos(betam*%pi/180))^3));
delp = eff_tt*psi*rho*Um^2;

//Results
printf('(i)The nominal deflection= %.1f deg.\n the nominal incidence = %.1f deg.',eps_,i_);
printf('\n (ii)The inlet flow angle, beta1 = alpha2 = %.1f deg\n outlet flow angle beta2 = alpha1 = %.1f deg.',beta1,beta2);
printf('\n (iii)The flow coefficient = %.3f\n The stage loading factor = %.3f',phi,psi);
```

```
printf('\n (iv) The rotor lift coefficient = %.2f.',
    CL);
printf('\n (v) The overall drag coefficient of each
    row = %.3f.',CD);
printf('\n (vi) The total-to-total stage efficiency
    = %.3f.\n The pressure rise across the stage = %d
     kPa',eff_tt,delp/1000);


//there are small errors in the answers given in
    textbook
```

Chapter 6

Three dimensional Flows in Axial Turbomachines

Scilab code Exa 6.1 Ex 1

```
clear;
clc;
funcprot(0);

//given data
dt = 1.0;//tip diameter in m
dh = 0.9;//hub diameter in m
alpha1 = 30;//in deg
beta1 = 60;//in deg
alpha2 = 60;//in deg
beta2 = 30;//in deg
N = 6000;//rotational speed in rev/min
rhog = 1.5;//gas density in kg/m^3
Rt = 0.5;//degree of reaction at the tip

//Calculations
omega = 2*%pi*N/60;
Ut = omega*0.5*dt;
Uh = omega*0.5*dh;
```

```
cx = Ut/(tan(alpha1*%pi/180) + tan(beta1*%pi/180));
mdot = %pi*((0.5*dt)^2 - (0.5*dh)^2)*rhog*cx;
Wcdot = mdot*Ut*cx*(tan(alpha2*%pi/180)- tan(alpha1*%pi/180));
ctheta1t = cx*tan(alpha1*%pi/180);
ctheta1h = ctheta1t*(dt/dh);
ctheta2t = cx*tan(alpha2*%pi/180);
ctheta2h = ctheta2t*(dt/dh);
alpha1_ = (180/%pi)*atan(ctheta1h/cx);
beta1_ = (180/%pi)*atan((Uh/cx) - tan(alpha1_*%pi/180));
alpha2_ = (180/%pi)*atan(ctheta2h/cx);
beta2_ = (180/%pi)*atan((Uh/cx) - tan(alpha2_*%pi/180));
k = Rt*(0.5*dt)^2;
Rh = 1 - (k/(0.5*dh)^2);

//Results
printf('(i)The axial velocity, cx = %d m/s',cx);
printf('\n (ii)The mass flow rate = %.1f kg/s',mdot);
printf('\n (iii)The power absorbed by the stage = %.1f MW',Wcdot/(10^6));
printf('\n (iv)The flow angles at the hub are:\n alpha1 = %.2f deg,\n beta1 = %.2f deg,\n alpha2 = %.1f deg, and\n beta2 = %.2f deg.',alpha1_,beta1_,alpha2_,beta2_);
printf('\n (v)The reaction ratio of the stage at the hub, R = %.3f.',Rh);


//there are small errors in the answers given in textbook
```

Scilab code Exa 6.2 Ex 2

```
clear;
clc;
funcprot(0);

//given data

R = 0.5;//degree of reaction
Cp = 1005;//kJ/(kgC)
cx1_Ut_rt = 0.4;
delT0 = 16.1;//temperature rise
Ut = 300;//in m/s

//calculations
A1 = cx1_Ut_rt^2 +(0.5-0.18*log(1));
c1 = 2*(1-R);
c2 = Cp*delT0/(2*Ut^2 *(1-R));
A2 = 0.56;
k = 0.4:0.01:1.0;
n = (1.0-0.4)/0.01 + 1;
i = 1;
for i = 1:n
    cx1_Ut(i) = sqrt(A1 - (c1^2)*(0.5*k(i)^2 - c2*
       log(k(i))));
    cx2_Ut(i) = sqrt(A2 - (c1^2)*(0.5*k(i)^2 + c2*
       log(k(i))));
    R_(i) = 0.778+log(k(i));
    Rn(i) = 0.5;
end

//Results
plot(k,cx1_Ut,'bo-');
plot(k,cx2_Ut,'<>r-');
title("Solution of exit axial-velocity profile for a
    first power stage","fontsize",3) ;//title of the
    plot
xlabel("Radius ratio, r/rt","fontsize",3) ;//x label
ylabel("cx/Ut","fontsize",3) ;//y label
legend(["(cx2/Ut)";"(cx1/Ut)"] , opt=2); //legend
```

```
    box
a=gca();
b = newaxes();
b.y_location = "right";
b.filled = "off";
b.axes_visible = ["off","on","on"];
b.axes_bounds = a.axes_bounds;
b.font_size = a.font_size;
plot(k,R_,"g");
plot(k,Rn,);
ylabel("Reaction","fontsize",3) ;//y label
legend(["True Reaction";"Nominal Reaction"] , opt=1)
    ; //legend box
```

Chapter 7

Centrifugal Pumps Fans and Compressors

Scilab code Exa 7.1 Ex 1

```
clear;
clc;
funcprot(0);

//function to calculate blade cavitation coefficient
function [res] = fun(sigmab,k,omega_ss)
    res = (sigmab^2)*(1 + sigmab)- (((3.42*k)^2)/(
        omega_ss^4));
endfunction

//given data
Q = 25;//flow rate in dm^3/s
omega = 1450;//rotational speed in rev/min
omega_ss = 3;//max. suction specific speed in rad/
    sec
r = 0.3;//inlet eye radius ratio
g = 9.81;//in m/s^2

//Calculations
```

```
k = 1-(r^2);
sigmab = 0.3;//initial guess
res = fun(sigmab,k,omega_ss);//initial value
i = 0;
while (abs(res)>0.0001)
    if res>0.0 then
        sigmab = sigmab - 0.0001;
    elseif res<0.0
        sigmab = sigmab + 0.0001;
    end
    res = fun(sigmab,k,omega_ss);
end
phi = (sigmab/(2*(1+sigmab)))^0.5;
rs1 = ((Q*10^-3)/(%pi*k*(omega*%pi/30)*phi))^(1/3);
ds1 = 2*rs1;
cx1 = phi*(omega*%pi/30)*rs1;
Hs = (0.75*sigmab*cx1^2)/(g*phi^2);

//Results
printf('(i)The blade cavitation coefficient = %.3f',
    sigmab);
printf('\n (ii)The shroud radius at the eye = %.5f m
    \n The required diameter of the eye = %.1f mm',
    rs1,ds1*10^3);
printf('\n (iii)The eye axial velocity = %.3f m/s',
    cx1);
printf('\n (iv)The NPSH = %.3f m',Hs);
```

Scilab code Exa 7.2 Ex 2

```
clear;
clc;
funcprot(0);

//given data
```

```
alpha1 = 30;//prewhirl in deg
hs = 0.4;//inlet hub-shrub radius ratio
Mmax = 0.9;//max Mach number
Q = 1;//air mass flow in kg/s
p01 = 101.3;//stagnation pressure in kPa
T01 = 288;//stagnation temperature in K
gamma = 1.4;
Rg = 287;//in J/(kgK)

//Calculations
beta1 = 49.4;//in deg
f = 0.4307;
a01 = sqrt(gamma*Rg*T01);
rho01 = p01*1000/(Rg*T01);
k = 1-(hs^2);
omega = (%pi*f*k*rho01*a01^3)^0.5;
N = (omega*60/(2*%pi));
rho1 = rho01/(1 + 0.2*(Mmax*cos(beta1*%pi/180))^2)^2.5;
cx = ((omega^2)/(%pi*k*rho1*(tan(beta1*%pi/180) + tan(alpha1*%pi/180))^2))^(1/3);
rs1 = (1/(%pi*rho1*cx*k))^0.5;

ds1 = 2*rs1;
U = omega*rs1;

//Results
printf('(i)The rotational speed of the impeller = %.1f rad/s and N = %d rev/min.',omega,N);
printf('\n (ii)The inlet static density downstream of the guide vanes at the shroud = %.5f kg/m^3.\n The axial velocity = %.2f m/s.',rho1,cx);
printf('\n (iii)The inducer tip diameter = %.3f cm\n U = %.1f m/s.',ds1*100,U);

//there are small errors in the answers given in textbook
```

Scilab code Exa 7.3 Ex 3

```
clear;
clc;
funcprot(0);

//given data
Q = 0.1;//in m^3/s
N = 1200;//rotational speed in rev/min
beta2_ = 50;//in deg
D = 0.4;//impeller external diameter in m
d = 0.2;//impeller internal diameter in m
b2 = 31.7;//axial width in mm
eff = 0.515;//diffuser efficiency
H = 0.1;//head losses
De = 0.15;//diffuser exit diameter
A = 0.77;
B = 1;
g = 9.81;

//Calculations
U2 = %pi*N*D/60;
cr2 = Q/(%pi*D*b2/1000);
sigmaB = (A - H*tan(beta2_*%pi/180))/(B - H*tan(
    beta2_*%pi/180));
ctheta2 = sigmaB*U2*(1-H*tan(beta2_*%pi/180));
Hi = U2*ctheta2/g;
c2 = sqrt(cr2^2 + ctheta2^2);
c3 = 4*Q/(%pi*De^2);
HL = 0.1*Hi + 0.485*((c2^2)-(c3^2))/(2*g) + (c3^2)
    /(2*g);
H = Hi - HL;
eff_hyd = H/Hi;

```

```
//Results
printf('The slip factor = %.3f.',sigmaB);
printf('\n The manometric head = %.1f m.',H);
printf('\n The hydraulic efficiency = %.1f
   percentage.',eff_hyd*100);

//there is a very small error in the answer given in
    textbook
```

Scilab code Exa 7.4 Ex 4

```
clear;
clc;
funcprot(0);

//given data
T01 = 22;//stagnation temperature in degC
Z = 17;//number of vanes
N = 15000;//rotational speed in rev/min
r = 4.2;//stagnation pressure ratio between diffuser
    and impeller
eff_ov = 0.83;//overall efficiency
mdot = 2;//mass flow rate in kg/s
eff_m = 0.97;//mechanical efficiency
rho2 = 2;//air density at impeller outle in kg/m^3
gamma = 1.4;
R = 0.287;//in kJ/(kg.K)
b2 = 11;//axial width at the entrance to the
   diffuser in mm

//Calculations
Cp = gamma*R*1000/(gamma-1);
sigmaS = 1 - 2/Z;
U2 = sqrt(Cp*(T01+273)*((r)^((gamma-1)/gamma) -1)/(
   sigmaS*eff_ov));
```

```
omega = N*%pi/30;
rt = U2/omega;
Wdot_act = mdot*sigmaS*(U2^2)/(eff_m);
cr2 = mdot/(rho2*2*%pi*rt*b2/1000);
ctheta2 = sigmaS*U2;
c2 = sqrt(ctheta2^2 +cr2^2);
delW = sigmaS*U2^2;
T2 = T01+273+(delW - 0.5*c2^2)/Cp;
M2 =  c2/sqrt(gamma*R*1000*T2);

//Results
printf('Absolute mach number, M2 = %.2f.',M2);
```

Chapter 8

Radial Flow Gas Turbines

Scilab code Exa 8.1 Ex 1

```
clear;
clc;
funcprot(0);

//given data
D2 = 23.76;//diameter of rotor in cm
N = 38140;//rotational speed in rev/min
alpha2 = 72;//absolute flow angle in deg
d = 0.5*D2;//rotor mean exit diameter

//Calcultaions
U2 = %pi*N*D2/(100*60);
w2 = U2/tan(alpha2*%pi/180);
c2 = U2*sin(alpha2*%pi/180);
w3 = 2*w2;
U3 = 0.5*U2;
c3 = sqrt(w3^2 - U3^2);
delW = 0.5*((U2^2 - U3^2)+(w3^2 - w2^2)+(c2^2 - c3
   ^2));
inp_U2 =  0.5*(U2^2 - U3^2)/delW;
inp_w2 = 0.5*(w3^2 - w2^2)/delW;
```

```
inp_c2 = 0.5*(c2^2 - c3^2)/delW;

//Results
printf('The fractional inputs from the three terms are, for the U^2 terms, %.3f; \n for the w^2 terms, %.3f; for the c^2 terms, %.3f.',inp_U2,inp_w2,inp_c2);

//there are errors in the answers given in textbook
```

Scilab code Exa 8.2 Ex 2

```
clear;
clc;
funcprot(0);

//given data
r = 1.5;//operating pressure ratio
K1 = 1.44*10^-5;
K2 = 2410;
K3 = 4.59*10^-6;
T01 = 400;//in K
D2 = 72.5;//rotor inlet diamete in mm
D3_av = 34.4;//rotor meaan outlet diameter in mm
b = 20.1;//rotor outlet annulus width in mm
zetaN = 0.065;//enthalpy loss coefficient
alpha2 = 71;//in deg
beta3_av = 53;//in deg
Cp = 1005;//inJ/(kg.K)
gamma = 1.4;

//Calculations
N = K2*sqrt(T01);
U2 = %pi*N*D2/(60*1000)
delW = U2^2;
```

```
delh = Cp*T01*(1-(1/r)^((gamma-1)/gamma));
eff_ts = delW/(delh);
delW_act = K3*K2*%pi*T01/(30*K1);
eff_ov = delW_act/delh;
zetaR = (2*((1/eff_ts)-1) - (zetaN/sin(alpha2*%pi/180)))*((D2/D3_av)^2)*(sin(beta3_av*%pi/180))^2 - (cos(beta3_av*%pi/180))^2;
r3 = 0.5*(D3_av-b)*10^-3;
w3_w2av_min = (D3_av/D2)*tan(alpha2*%pi/180)*((2*r3/D3_av)^2 + (1/tan(beta3_av*%pi/180))^2)^0.5;
w3_w2av = (D3_av/D2)*tan(alpha2*%pi/180)*(1+((1/tan(beta3_av*%pi/180))^2))^0.5;

//Results
printf('The total-to-static efficiency = %.2f percentage.',eff_ts*100);
printf('\n The overall efficiency = %.2f percentage.',eff_ov*100);
printf('\n The rotor enthalpy loss coefficient = %.3f',zetaR);
printf('\n The rotor relative velocity ratio = %.2f',w3_w2av);


//there are small errors in the answers given in textbook
```

Scilab code Exa 8.3 Ex 3

```
clear;
clc;
funcprot(0);

//given data
Z = 12;//number of vanes
```

```
delW = 230;//in kW
T01 = 1050;//stagnation temperature in K
mdot = 1;//flow rate in kg/s
eff_ts = 0.81;//total-to-static efficiency
Cp = 1.1502;//in kJ/(kg.K)
gamma = 1.333;
R = 287;//gas constant

//Calculations
S = delW/(Cp*T01);
alpha2 = (180/%pi)*acos(sqrt(1/Z));
beta2 = 2*(90-alpha2);
p3_p01 = (1-(S/eff_ts))^(gamma/(1-gamma));
M02 = sqrt((S/(gamma-1))*((2*cos(beta2*%pi/180))/(1+cos(beta2*%pi/180))));
M2 = sqrt((M02^2)/(1-0.5*(gamma-1)*(M02^2)));
U2 = sqrt((gamma*R*T01)*(1/cos(beta2*%pi/180))*(S/(gamma-1)));

//Results
printf('(i) The absolut and relative flow angles:\n alpha2 = %.2f deg\n beta2 = %.2f deg',alpha2,beta2);
printf('\n (ii) The overall pressure ratio = %.3f',p3_p01);
printf('\n (iii) The rotor rip speed = %.1f m/s\n The inlet absolute Mach number = %.3f',U2,M2);


//there are small errors in the answers given in textbook
```

Scilab code Exa 8.4 Ex 4

```
clear;
```

```
clc;
funcprot(0);

//given data
cm3_U2 = 0.25;
nu = 0.4;
r3s_r2 = 0.7;
w3av_w2 = 2.0;

//Calculations
r3av_r3s = 0.5*(1+nu);
r3av_r2 = r3av_r3s*r3s_r2;
beta3_av = (180/%pi)*atan(r3av_r2/cm3_U2);
beta3s = (180/%pi)*atan(r3s_r2/cm3_U2);
w3s_w2 = 2*cos(beta3_av*%pi/180)/cos(beta3s*%pi/180)
    ;

//Results
printf('The relative velocity ratio = %.3f.',w3s_w2)
    ;
```

Scilab code Exa 8.5 Ex 5

```
clear;
clc;
funcprot(0);

//given data
Z = 12;//number of vanes
delW = 230;//in kW
T01 = 1050;//stagnation temperature in K
mdot = 1;//flow rate in kg/s
eff_ts = 0.81;//total-to-static efficiency
Cp = 1.1502;//in kJ/(kg.K)
gamma = 1.333;
```

```
R = 287;//gas constant
cm3_U2 = 0.25;
nu = 0.4;
r3s_r2 = 0.7;
w3av_w2 = 2.0;
p3 = 100;//static pressure at rotor exit in kPa
zetaN = 0.06;//nozzle enthalpy loss coefficient
U2 = 538.1;//in m/s
p01 = 3.109*10^5;//in Pa

//Calculations
S = delW/(Cp*T01);
T03 = T01*(1-S);
T3 = T03 - (cm3_U2^2)*(U2^2)/(2*Cp*1000);
r2 = sqrt(mdot/((p3*1000/(R*T3))*(cm3_U2)*U2*%pi*(r3s_r2^2)*(1-nu^2)));
D2 = 2*r2;
omega = U2/r2;
N = omega*30/%pi;
ctheta2 = S*Cp*1000*T01/U2;
alpha2 = (180/%pi)*acos(sqrt(1/Z));
cm2 = ctheta2/tan(alpha2*%pi/180);
c2 = ctheta2/sin(alpha2*%pi/180);
T2 = T01 - (c2^2)/(2*Cp*1000);
p2 = p01*(1-(((c2^2)*(1+zetaN))/(2*Cp*1000*T01)))^(gamma/(gamma-1));
b2_D2 = (0.25/%pi)*(R*T2/p2)*(mdot/(cm2*r2^2));

//Results
printf('(i) The diamaeter of the rotor = %.4f m\n its speed of rotation = %.1f rad/s (N = %d rev/min)',D2,omega,N);
printf('\n(ii) The vane width to diameter ratio at rotor inlet = %.4f',b2_D2);

//there are some errors in the answers given in textbook
```

Scilab code Exa 8.6 Ex 6

```
clear;
clc;
funcprot(0);

//given data
Z = 12;//number of vanes
delW = 230;//in kW
T01 = 1050;//stagnation temperature in K
mdot = 1;//flow rate in kg/s
eff_ts = 0.81;//total-to-static efficiency
Cp = 1.1502;//in kJ/(kg.K)
gamma = 1.333;
R = 287;//gas constant
cm3_U2 = 0.25;
nu = 0.4;
r3s_r2 = 0.7;
w3av_w2 = 2.0;
p3 = 100;//static pressure at rotor exit in kPa
zetaN = 0.06;//nozzle enthalpy loss coefficient
U2 = 538.1;//in m/s
p01 = 3.109*10^5;//in Pa

//results of Example 8.4 and Example 8.5
r3av_r3s = 0.5*(1+nu);
r3av_r2 = r3av_r3s*r3s_r2;
alpha2 = (180/%pi)*acos(sqrt(1/Z));
beta2 = 2*(90-alpha2);
beta3_av = (180/%pi)*atan(r3av_r2/cm3_U2);
beta3s = (180/%pi)*atan(r3s_r2/cm3_U2);
w3s_w2 = 2*cos(beta3_av*%pi/180)/cos(beta3s*%pi/180)
    ;
S = delW/(Cp*T01);
```

```
T03 = T01*(1-S);
T3 = T03 - (cm3_U2^2)*(U2^2)/(2*Cp*1000);
r2 = sqrt(mdot/((p3*1000/(R*T3))*(cm3_U2)*U2*%pi*(r3s_r2^2)*(1-nu^2)));
D2 = 2*r2;
omega = U2/r2;
N = omega*30/%pi;
ctheta2 = S*Cp*1000*T01/U2;
alpha2 = (180/%pi)*acos(sqrt(1/Z));
cm2 = ctheta2/tan(alpha2*%pi/180);
c2 = ctheta2/sin(alpha2*%pi/180);
T2 = T01 - (c2^2)/(2*Cp*1000);
p2 = p01*(1-(((c2^2)*(1+zetaN))/(2*Cp*1000*T01)))^(gamma/(gamma-1));
b2_D2 = (0.25/%pi)*(R*T2/p2)*(mdot/(cm2*r2^2));

//Calculations
c3 = cm3_U2*U2;
cm3 = c3;
w3_av = 2*cm3/(cos(beta2*%pi/180));
w2 = w3_av/2;
c0 = sqrt(2*delW*1000/eff_ts);
zetaR = (c0^2 *(1-eff_ts)- (c3^2)- zetaN*(c2^2))/(w3_av^2);
i = beta2;
n = 1.75;
eff_ts_new = 1-((c3^2)+zetaN*(c2^2)+zetaR*(w3_av^2)+(1-(cos(i*%pi/180))^n)*(w2^2))/(c0^2);

//Results
printf('(a)The rotor enthalpy loss coefficient = %.4f',zetaR);
printf('\n(b) The total-to-static efficiency of the turbine = %.3f',eff_ts_new);


//there are some errors in the answers given in textbook
```

Chapter 9

Hydraulic Turbines

Scilab code Exa 9.1 Ex 1

```
clear;
clc;
funcprot(0);

//given data
Q = 2.272;//water volume flow rate in m^3/s
l = 300;//length in m
Hf = 20;//head loss in m
f = 0.01;//friction factor
g = 9.81;//acceleration due to gravity in m/s^2

//Calculations
d = (32*f*l*((Q/%pi)^2)/(g*Hf))^(1/5);

//Results
printf('The diameter of the pipe = %.4f m',d);
```

Scilab code Exa 9.2 Ex 2

```
clear;
clc;
funcprot(0);

//given data
P = 4.0;//in MW
N = 375;//in rev/min
H_eps = 200;//in m
KN = 0.98;//nozzle velocity coefficient
d = 1.5;//in m
k = 0.15;//decrease in relative flow velocity across the buckets
alpha = 165;//in deg
g = 9.81;//in m/s^2
rho = 1000;//in kg/m^3

//Calculations
U = N*%pi*d*0.5/30;
c1 = KN*sqrt(2*g*H_eps);
nu = U/c1;
eff = 2*nu*(1-nu)*(1-(1-k)*cos(alpha*%pi/180));
Q = (P*10^6 /eff)/(rho*g*H_eps);
Aj = Q/(2*c1);
dj = sqrt(4*Aj/%pi);
omega_sp = (N*%pi/30)*sqrt((P*10^6)/rho)/((g*H_eps)^(5/4));

//Results
printf('(i)The runner efficiency = %.4f',eff);
printf('\n (ii)The diameter of each jet = %.4f m',dj);
printf('\n (iii)The power specific speed = %.3f rad',omega_sp);
```

Scilab code Exa 9.3 Ex 3

```
clear;
clc;
funcprot(0);

//given data
H_eps = 150;//in m
z = 2;//in m
U2 = 35;//runner tip speed in m/s
c3 = 10.5;//meridonal velocity of water in m/s
c4 = 3.5;//velocity at exit in m/s
delHN = 6.0;//in m
delHR = 10.0;//in m
delHDT = 1.0;//in m
g = 9.81;//in m/s^2
Q = 20;//in m^3/s
omega_sp = 0.8;//specific speed of turbine in rad
c2 = 38.73;//in m/s

//Calculations
H3 = ((c4^2 - c3^2)/(2*g)) + delHDT - z;
H2 = H_eps-delHN-(c2^2)/(2*g);
delW = g*(H_eps-delHN-delHR-z)-0.5*c3^2 -g*H3;
ctheta2 = delW/U2;
alpha2 = (180/%pi)*atan(ctheta2/c3);
beta2 = (180/%pi)*atan((ctheta2-U2)/c3);
eff_H = delW/(g*H_eps);
omega = (omega_sp*(g*H_eps)^(5/4))/sqrt(Q*delW);
N = omega*30/%pi;
D2 = 2*U2/omega;

//Results
printf('(i)The pressure head H3 relative to the trailrace = %.1f m\n The pressure head H2 at exit from the runner = %.2f m',H3,H2);
printf('\n(ii)The flow angles at runner inlet and at guide vane exit:\n alpha2 = %.1f deg\n beta2 = %.2f deg',alpha2,beta2);
printf('\n(iii)The hydraulic efficiency of the
```

```
    turbine = %.4f',eff_H);
printf('\n The speed of rotation , N = %d rev/min',N)
    ;
printf('\n The runner diameter is , D2 = %.3f m',D2);


//there are small errors in the answers given in
    textbook
```

Scilab code Exa 9.4 Ex 4

```
clear;
clc;
funcprot(0);

//function to calculate flow angles
function [alpha2,beta2,beta3] =fun(r,N,cx2,ctheta2)
    alpha2 = (180/%pi)*atan(ctheta2/cx2);
    beta2 = (180/%pi)*atan((U2)*(r)/cx2 - tan(alpha2
        *%pi/180));
    beta3 = (180/%pi)*atan((U2)*r/cx2) ;
endfunction

//given data
P = 8;//output power in MW
HE = 13.4;//available head at entry in m
N = 200;//in rev/min
L = 1.6;//length of inlet guide vanes
d1 = 3.1;//diameter of trailing edge in m
D2t = 2.9;//runner diameter in m
nu = 0.4;//hub-tip ratio
eff = 0.92;//hydraulic efficiency
rho = 1000;//density in kg/m^3
g = 9.81;//acceleration due to gravity in m/s^2

```

```
//Calculations
Q = P*10^6 /(eff*rho*g*HE);
cr1 = Q/(2*%pi*0.5*d1*L);
cx2 = 4*Q/(%pi*D2t^2 *(1-nu^2));
U2 = N*(%pi/30)*D2t/2;
ctheta2 = eff*g*HE/U2;
ctheta1 = ctheta2*(D2t/d1);
alpha1 = (180/%pi)*atan(ctheta1/cr1);

//calculating flow angle for diffrent radii
[alpha21,beta21,beta31] = fun(1.0,U2,cx2,ctheta2);
[alpha22,beta22,beta32] = fun(0.7,U2,cx2,ctheta2/0.7);
[alpha23,beta23,beta33] = fun(0.4,U2,cx2,ctheta2/0.4);

//Results
printf('Calculated values of flow angles:\n Parameter                              Ratio of r/ri                  ');
printf('\n---------------------------------------------------------------------------');
printf('\n                                   0.4                0.7                1.0 ');
printf('\n----------------------------------------------');
printf('\n ctheta2(in m/s)        %.3f              %.3f              %.3f',ctheta2/0.4,ctheta2/0.7,ctheta2/1.0);
printf('\n tan(alpha2)            %.3f              %.4f             %.3f',tan(alpha23*%pi/180),tan(alpha22*%pi/180),tan(alpha21*%pi/180));
printf('\n alpha2(deg)            %.2f              %.2f              %.2f',alpha23,alpha22,alpha21);
printf('\n U/cx2                  %.3f              %.4f             %.3f',(U2/cx2)*0.4,(U2/cx2)*0.7,(U2/cx2)*1.0);
```

```
printf('\n beta2(deg)            %.2f            %.2f
            %.2f',beta23,beta22,beta21);
printf('\n beta3(deg)            %.2f            %.2f
            %.2f',beta33,beta32,beta31);
printf('\n
    ----------------------------------------------------------------------------
    ');
```

Scilab code Exa 9.5 Ex 5

```
clear;
clc;
funcprot(0);

//given data
k = 1/5;//scale ratio
Pm = 3;//in kW
Hm = 1.8;//in m
Nm = 360;//in rev/min
Qm = 0.215;//in m^3/s
Hp = 60;//in m
n = 0.25;
rho = 1000;//in kg/m^3
g = 9.81;//in m/s^2

//Calculations
Np = Nm*k*(Hp/Hm)^0.5;
Qp = Qm*(Nm/Np)*(1/k)^3;
Pp = Pm*((Np/Nm)^3)*(1/k)^5;
eff_m = Pm*1000/(rho*Qm*g*Hm);
eff_p = 1 - (1-eff_m)*0.2^n;
Pp_corrected = Pp*eff_p/eff_m;

//Results
printf('The speed = %.1f rev/min.',Np);
```

```
printf('\n The flow rate = %.2f m^3/s.',Qp);
printf('\n Power of the full-scale = %.2f MW.',Pp/1000);
printf('\n The efficiency of the model turbine = %.2f.',eff_m);
printf('\n The efficiency of the prototype = %.4f.',eff_p);
printf('\n The power of the full-size turbine = %.1f MW.',Pp_corrected/1000);
```

Scilab code Exa 9.6 Ex 6

```
clear;
clc;
funcprot(0);

//given data
//data from EXAMPLE 9.3
H_eps = 150;//in m
z = 2;//in m
U2 = 35;//runner tip speed in m/s
c3 = 10.5;//meridonal velocity of water in m/s
c4 = 3.5;//velocity at exit in m/s
delHN = 6.0;//in m
delHR = 10.0;//in m
delHDT = 1.0;//in m
g = 9.81;//in m/s^2
Q = 20;//in m^3/s
omega_sp = 0.8;//specific speed of turbine in rad
c2 = 38.73;//in m/s

//data from this example
Pa = 1.013;//atmospheric pressure in bar
Tw = 25;//temperature of water in degC
Pv = 0.03166;//vapor pressure of water at Tw
```

```
rho = 1000;//density of wate in kg/m^3
g = 9.81;//acceleration due to gravity in m/s^2

H3 = ((c4^2 - c3^2)/(2*g)) + delHDT - z;
H2 = H_eps-delHN-(c2^2)/(2*g);
delW = g*(H_eps-delHN-delHR-z)-0.5*c3^2 -g*H3;
ctheta2 = delW/U2;
alpha2 = (180/%pi)*atan(ctheta2/c3);
beta2 = (180/%pi)*atan((ctheta2-U2)/c3);
eff_H = delW/(g*H_eps);
omega = (omega_sp*(g*H_eps)^(5/4))/sqrt(Q*delW);

Hs = (Pa-Pv)*(10^5)/(rho*g) - z;
sigma = Hs/H_eps;
omega_ss = omega*(Q^0.5)/(g*Hs)^(3/4);

//Results
printf('The NSPH for the turbine = %.3f m.',Hs);
if omega_ss>4.0 then
    printf('\n Since the suction specific speed (= %
       .4f.)is greater than 4.0(rad), the cavitation
        is likely to occur.',omega_ss);
end

//there is small error in the answer given in
   textbook
```

Chapter 10

Wind Turbines

Scilab code Exa 10.2 Ex 2

```
clear;
clc;
funcprot(0);

//given data
a_ = 1/3;

//Calculations
R2_R1 = 1/(1-a_)^0.5;
R3_R1 = 1/(1-2*a_)^0.5;
R3_R2 = ((1-a_)/(1-2*a_))^0.5;

//Results
printf('R2/R1 = %.3f\n R3/R1 = %.3f\n R3/R2 = %.3f',
    R2_R1,R3_R1,R3_R2);
```

Scilab code Exa 10.3 Ex 3

```
clear;
clc;
funcprot(0);

//given data
d = 30;//tip diameter in m
cx1 = 7.5;//in m/s
cx2 = 10;//in m/s
rho = 1.2;//in kg/m^3
a_ = 1/3;

//Calculations
P1 = 2*a_*rho*(%pi*0.25*d^2)*(cx1^3)*(1-a_)^2;
P2 = 2*a_*rho*(%pi*0.25*d^2)*(cx2^3)*(1-a_)^2;


//Results
printf('(i)With cx1 = %.1f m/s, P = %d kW.',cx1,P1
    /1000);
printf('\n(ii)With cx1 = %d m/s, P = %.1f kW.',cx2,
    P2/1000);
```

Scilab code Exa 10.4 Ex 4

```
clear;
clc;
funcprot(0);

//given data
P = 20;//power required in kW
cx1 = 7.5;//steady wind speed in m/s
rho = 1.2;//density in kg/m^3
Cp = 0.35;
eta_g = 0.75;//output electrical power
eff_d = 0.85;//electrical generation efficiency
```

```
//Calculations
A2 = 2*P*1000/(rho*Cp*eta_g*eff_d*cx1^3);
D2 = sqrt(4*A2/%pi);

//Results
printf('The diameter = %.1f m.',D2);
```

Scilab code Exa 10.5 Ex 5

```
clear;
clc;
funcprot(0);

//given data
Z = 3;//number of blades
D = 30;//rotor diameter in m
J = 5.0;//tip-speed ratio
l = 1.0;//blade chord in m
r_R = 0.9;//ratio
beta = 2;//pitch angle in deg

//Calculations
//iterating to get values of induction factors
a = 0.0001;//inital guess
a_ = 0.0001;//inital guess
a_new = 0.0002;//inital guess
i = 0;
while (a_~=a_new)
   phi = (180/%pi)*atan((1/(r_R*J))*((1-a)/(1-a_)));
   alpha = phi-beta;
   CL = 0.1*alpha;
   lamda = (Z*l*CL)/(8*%pi*0.5*r_R*D);
   a = 1/(1+(1/lamda)*sin(phi*%pi/180)*tan(phi*%pi
      /180));
```

```
    a_new = 1/((1/lamda)*cos(phi*%pi/180) -1);
    if a_ < a_new
        a_ = a_ + 0.0001;
    elseif a_ > a_new
        a_ = a_ - 0.0001;
    end
    if (abs((a_-a_new)/a_new) < 0.1) then
        break;
    end
    i = i+1;
end

//Results
printf('Axial induction factor, a = %.4f',a);
printf('\n Tangential induction factor = %.5f',a_new);
printf('\n phi = %.3f deg.',phi);
printf('\n Lift coefficient = %.3f.',CL);

//The answers given in textbook are wrong
```

Scilab code Exa 10.6 Ex 6

```
clear;
clc;
funcprot(0);

//given data
D = 30;//tip diameter in m
CL = 0.8;//lift coefficient
J = 5.0;
l = 1.0;//chord length in m
Z = 3;//number of blades
r_R = [0.2 0.3 0.4 0.6 0.8 0.9 0.95 1.0];
n = 8;
```

```
//Calculations
//iterating to get values of induction factors
a = 0.1;//inital guess
anew = 0;
a_ = 0.006;//inital guess
a_new = 0.0;//inital guess
for i = 1:n
    while (a_~=a_new)
        lamda = (Z*l*CL)/(8*%pi*0.5*r_R(i)*D);
        phi = (180/%pi)*atan((1/(r_R(i)*J))*((1-a)/(1-a_)));
        a = 1/(1+(1/lamda)*sin(phi*%pi/180)*tan(phi*%pi/180));
        a_new = 1/((1/lamda)*cos(phi*%pi/180) -1);
        alpha = CL/0.1;
        beta = phi-alpha;
        if a_ < a_new
            a_ = a_ + 0.0001;
        elseif a_ > a_new
            a_ = a_ - 0.0001;
        end
        if (abs((a_-a_new)/a_new) < 0.01) then
            break;
        end
    end
    p(i) = phi;b(i) = beta;a1(i) = a;a2(i) = a_new;
end

//Results
printf('Summary of results of iterations (N.B. CL = 0.8 along the span)');
printf('\n --------------------------------------------------------------------------------');
printf('\n r/R    %.1f    %.1f    %.1f    %.1f    %.1f    %.1f    %.2f    %.1f ',r_R(1),r_R(2),r_R(3),r_R(4),r_R(5),r_R(6),r_R(7),r_R(8));
```

```
printf('\n ---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------');
printf('\n phi     %.2f       %.2f       %.2f       %.2f       %.2f      %.2f       %.2f          %.3f',p(1),p(2),p(3),p(4),p(5),p(6),p(7),p(8));
printf('\n beta    %.2f       %.2f       %.2f       %.2f       %.2f       %.2f         %.2f          %.2f',b(1),b(2),b(3),b(4),b(5),b(6),b(7),b(8));
printf('\n a       %.4f    %.5f    %.5f    %.4f    %.4f    %.4f       %.4f       %.4f',a1(1),a1(2),a1(3),a1(4),a1(5),a1(6),a1(7),a1(8));
printf('\n a`      %.5f    %.5f    %.5f    %.5f    %.5f    %.5f    %.5f    %.5f',a2(1),a2(2),a2(3),a2(4),a2(5),a2(6),a2(7),a2(8));
printf('\n ---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------');

//there are some errors in the answers given in textbook
```

Scilab code Exa 10.7 Ex 7

```
clear;
clc;
funcprot(0);

//given data
//data from Exampla 10.5
Z = 3;//number of blades
D = 30;//rotor diameter in m
J = 5.0;//tip-speed ratio
l = 1.0;//blade chord in m
beta = 2;//pitch angle in deg
```

```
omega = 2.5;//in rad/s

rho = 1.2;//density in kg/m^3
cx1 = 7.5;//in m/s
sum_var1 = 6.9682;//from Table 10.3
sum_var2 = 47.509*10^-3;//from Table 10.4

//Calculations
X = sum_var1*0.5*rho*Z*l*0.5*D*cx1^2;
tau = sum_var2*0.5*rho*Z*l*(omega^2)*(0.5*D)^4;
P = tau*omega;
A2 = 0.25*%pi*D^2;
P0 = 0.5*rho*A2*cx1^3;
Cp = P/P0;
zeta = (27/16)*Cp;

//Results
printf('The total axial force = %d N.',X);
printf('\n The torque = %.3f *10^3 Nm.',tau/1000);
printf('\n The power developed = %.3f kW.',P/1000);
printf('\n The power coefficient = %.3f',Cp);
printf('\n The relative power coefficient = %.3f',
    zeta);
```

Scilab code Exa 10.8 Ex 8

```
clear;
clc;
funcprot(0);

//given data
X = 10583;//in N
D = 30;//rotor diameter in m
Cx = X/23856;
rho = 1.2;//density in kg/m^3
```

```
cx1 = 7.5;//in m/s

//sloving quadratic eqaution
a = 0;//inital guess
res = 1;
i = 0;
while (res~=0)
    res = a*(1-a) - Cx/4;
    if (res>0) then
        a = a-0.001;
    elseif (res<0)
        a = a+0.001;
    end
    if abs(res)<0.0001
        break;
    end
end
A2 = 0.25*%pi*D^2
P = 2*rho*A2*(cx1^3)*a*(1-a)^2;

//Results
printf('P = %.3f kW.',P/1000);

//there is small error in the answer given in
   textbook
```

Scilab code Exa 10.9 Ex 9

```
clear;
clc;
funcprot(0);

//given data
//data from Exampla 10.5
Z = 3;//number of blades
```

```
D = 30;//rotor diameter in m
J = 5.0;//tip-speed ratio
l = 1.0;//blade chord in m
beta = 1.59;//pitch angle in deg
omega = 2.5;//in rad/s
rho = 1.2;//density in kg/m^3
cx1 = 7.5;//in m/s
c1 = 1518.8;//from Ex 10.6
c2 = 0.5695*10^6;
P0 = 178.96;//Power developed in kW from Ex 10.7
X1 = 10582;//Total axial force in N from Ex 10.7
Cp1 = 0.378;//Power coefficient from Ex 10.7
zeta1 = 0.638;//rekative power coefficient from Ex 10.7

//Calculations
r_R = 0.25:0.1:0.95;
b = [28.4;19.49;13.80;9.90;7.017;4.900;3.00;1.59];
//b = [27.2985;17.8137;11.8231;7.8176;4.9972;3.0511;1.6476;1.59];
for j = 1:8
    i = 1;
    atemp = 0; a_temp = 0;
    while i>0,        i = i+1;
        f = (2/%pi)*acos(exp(-0.5*Z*(1-r_R(j))*(1+J^2)^0.5));
        phi = (180/%pi)*atan((1/(J*r_R(j)))*((1-atemp)/(1+a_temp)));
        CL = (phi-b(j))/10;
        lamda = f/(63.32/CL);
        anew = (lamda*cos(phi*%pi/180)/(lamda*cos(phi*%pi/180)+f*(sin(phi*%pi/180))^2));
        if atemp<anew then
            atemp = atemp+0.0001;
        elseif atemp>anew
            atemp = atemp-0.0001;
        end
```

```
            if (abs((atemp-anew)/anew) < 0.001) then
                break;
            end
        end
        F(j) = f;
        ph(j) = phi;
        cl(j) = CL;
        a(j) = anew;
        Var1(j) = ((1-anew)/sin(phi*%pi/180))^2 *cos(phi
           *%pi/180)*CL*0.1;
//      a_(j) = lamda/(F*cos(phi*%pi/180)-lamda);
//printf('r_R = %.2f, F = %.4f, a = %.4f, phi = %.4f
   \n',r_R(j),F(j),a(j),ph(j));
end

for k = 1:8
    lam(k) = F(k)*cl(k)/63.32;
    a_new(k) = lam(k)/(F(k)*cos(ph(k)*%pi/180)-lam(k
       ));
    Var2(k) = ((1+a_new(k))/cos(phi*%pi/180))^2 *(
       r_R(k))^3 *cl(k)*sin(ph(k)*%pi/180)*0.1;
end
X = c1*sum(Var1(1:8));
sum_Var2 = 40.707*10^-3;
tau = c2*sum(Var2(1:8));
P = tau*omega;
Cp = P/(P0*1000);
zeta = (26/17)*Cp;

//Results
printf('                    Summary of Results:');
printf('\n
   ------------------------------------------------------------------------
   ');
printf('\n                              Axial force,
   kN          Power, kW              Cp
   zeta');
printf('\n
```

```
---------------------------------------------------------------------------');
printf('\n Without tip correction              %.3f          %.2f                 %.3f              %.3f',X1/1000,P0*Cp1,Cp1,zeta1);
printf('\n With tip correction                 %.3f          %.2f                 %.3f              %.3f',X/1000,P/1000,Cp,zeta);
printf('\n ---------------------------------------------------------------------------');

//There are errors in the answers given in textbook
```

Scilab code Exa 10.10 Ex 10

```
clear;
clc;
funcprot(0);

//function to calculate values of blade chord and radius (optimum conditions)
function [j,lamda,r,l] = fun(phi)
    lamda = 1-cos(phi*%pi/180);
    j = sin(phi*%pi/180)*(2*cos(phi*%pi/180)-1)/(1+2*cos(phi*%pi/180))/(lamda);
    r = 3*j;
    l = 8*%pi*j*lamda;
endfunction

//given data
D = 30;//tip diameter in m
J = 5.0;//tip-speed ratio
Z = 3;//in m
CL = 1.0;
```

```

//Calculations
phi1 = 30;//in deg
phi2 = 20;//in deg
phi3 = 15;//in deg
phi4 = 10;//in deg
phi5 = 7.556;//in deg
//Values of blade chord and radius (optimum conditions)
[j1,lamda1,r1,l1] = fun(phi1);
[j2,lamda2,r2,l2] = fun(phi2);
[j3,lamda3,r3,l3] = fun(phi3);
[j4,lamda4,r4,l4] = fun(phi4);
[j5,lamda5,r5,l5] = fun(phi5);

printf('Values of blade chord and radius(optimum conditions):');
printf('\n ------------------------------------------------------------------------------ ');
printf('\n phi(deg)       j              4flamda              r(m)            l(m)');
printf('\n ------------------------------------------------------------------------------ ');
printf('\n %d          %.2f                 %.3f              %.1f               %.3f',phi1,j1,4*j1*lamda1,r1,l1);
printf('\n %d          %.2f                 %.3f              %.2f              %.3f',phi2,j2,4*j2*lamda2,r2,l2);
printf('\n %d          %.2f                 %.3f              %.2f              %.3f',phi3,j3,4*j3*lamda3,r3,l3);
printf('\n %d          %.3f                 %.4f             %.1f               %.3f',phi4,j4,4*j4*lamda4,r4,l4);
printf('\n %.3f       %d                   %.4f              %d                %.3f',phi5,ceil(j5),4*j5*lamda5,ceil(r5),l5);
printf('\n
```

```
    ____________________________________________________________________________
    ');

l_R = [l1,l2,l3,l4,l5]/(0.5*D);
r_R = [r1,r2,r3,r4,r5]/(0.5*D);
plot(r_R,l_R);
xlabel("r/R",'fontsize',3);
ylabel("l/R",'fontsize',3);
title("Optimal variation of chord length with radius
    ",'fontsize',3);

//there are very small errors in the ansers given in
    textbook
```