Scilab Textbook Companion for Advanced Engineering Fluid Mechanics by K. Muralidhar and G. Biswas[1]

Created by
Saurav Suman
B.Tech
Others
National Institute of Technology, Jamshedpur
College Teacher
None
Cross-Checked by
Spandana

June 7, 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: Advanced Engineering Fluid Mechanics

Author: K. Muralidhar and G. Biswas

Publisher: Alpha Science International

Edition: 2

Year: 2011

ISBN: 9788173196270

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

Derivation of Equations Governing Fluid Motion

Scilab code Exa 2.5 Pressure Drop

```
//Example 2.5
//Pressure Drop
//Page No. 56
clc;clear;close;

D=50;     //in mm
D=D/1000; //converted mm to m
u=5;        //in m/s^-1
L=6;     //in m
rho=995.6;      //in kg/m^-3
mu=79.77*10^-5;      //Pa-s
Re=(rho*u*D)/mu;
f=0.316/Re^(1/4);
dp=(f*L*u^2*rho)/(2*D);
m=rho*u*%pi*(D/2)^2;
P=m*dp/rho;
printf('\nPressure Drop = %f N/m^2\nPower Required =
    %f\n\n\n\nNote : Calculation mistakes in book.\
   nDifference in answer due to approximation in
```

```
book',dp,P);
```

Scilab code Exa 2.6 Pressure Drop

```
//Example 2.6
//Pressure Drop
//Page No. 56
clc;clear;close;

D=50;          //in mm
D=D/1000;  //converted mm to m
L=6;    //in m
rho=995.6;         //in kg/m^3
g=9.81;            //in m/s^-2
mu=79.77*10^-5;        //in Pa-s

//case 1
u=10;          //in m/s^-1
Re=(rho*u*D)/mu;
f=0.316/Re^(1/4);
dp(1)=(f*L*u^2*rho)/(2*D);


//case 2
u=20;                //in m/s^-1
Re=(rho*u*D)/mu;
f=0.316/Re^(1/4);
dp(2)=(f*L*u^2*rho)/(2*D);

u1=[10 20];      //in m
h_l=[dp(1)/(rho*g) dp(2)/(rho*g)];
f=interpln([h_l;u1],20)*%pi*D^2/4;
f=f*1000;              //conversion to lit/s
printf('\nFlow Rate = %g lit/s\n\nNote: Slight
   calculation errors in the book',f);
```

Chapter 5

Laminar Boundary Layers

Scilab code Exa 5.1 Skin Friction Drag

```
//Example 5.1
//Skin Friction Drag
//Page No. 288
clc;clear;close;

u=3;               //in m/s
L=1;               //in m
b=2;               //in m
rho=1.23;           //in kg/m^3
mu=1.46*10^-5;         //in m^2/s
Re=(u*L)/mu;
Cf=1.328/Re^(1/2);
F=Cf*rho*u*u*L*b/2;
d1=L*5/(Re)^(1/2);
d2=L*1.7208/(Re)^(1/2);
d3=L*0.664/(Re)^(1/2);
d1=d1*1000;         //conversion to mm
d2=d2*1000;         //conversion to mm
d3=d3*1000;         //conversion to mm
printf('\nDrag on the plate = %f N\nBoundary Layer
   Thickness = %f mm\nDisplacement Thickness = %f mm
```

```
\nMomentum Thickness = %f mm\n\n\n\n',F,d1,d2,d3)
;
```

Scilab code Exa 5.2 Maximum Bending Moment

```
//Example 5.2
//Maximum Bending Moment
//Page No. 290
clc;clear;close;

u=1.5;          //in m/s
L=6;             //in m
A=.15;           //in m
rho=1000;        //in kg/m^3
mu=1.02*10^-6;        //in m^2/s
Cd=2.1;
Re=(u*A)/mu;
F=Cd*rho*u*u*L*A/2;
Mo=F*L/2;
I=(A^4)/12;
sigma=Mo*(A/2)/I;
sigma=sigma/1000;       //conversion to kN/m^2 from N
    /m^2
printf('\nDrag = %f N\nBending Moment at the Base =
    %f Nm\nBending Stress at the Bottom = %f kN/m^2\n
    \n',F,Mo,sigma);
```

Scilab code Exa 5.5 Velocity Boundary Layer

```
//Example 5.5
//Velocity Boundary Layer
//Page No. 294
clc;clear;close;
```

```
U=2;            //in m/s
L=0.1;               //in m
x=0.05;            //in m
y=0.000225;              //in m
rho=983.1;           //in kg/m^3
f=0.629;         //refer book table 5.1;
mu=0.4748*10^-6;     //m^2/s
Re=(U*x)/mu;
d1=x*5/(Re)^(1/2);
n=y*(U/(mu*x))^(1/2);
u=U*f;
Re_L=U*L/mu;
Cf=1.328/Re_L^(1/2);
F=Cf*rho*U*U*L/2;
printf('\nThickness of velocity boundary layer at x
   = 5cm = %f m\nFluid Velocity at y = 0.0225cm = %f
    m/s\nDrag = %f N\n\n',d1,u,F);
```

Scilab code Exa 5.6 Velocity Boundary Layer

```
//Example 5.6
//Velocity Boundary Layer
//Page No. 295
clc;clear;close;

U=2;                   //in m/s
L=0.1;                   //in m
x=0.05;                 //in m
y=0.000225;              //in m
rho=983.1;               //kg/m^3
mu=0.4748*10^-6;           //m^2/s
Re=(U*x)/mu;
d1=x*4.64/(Re)^(1/2);
n=y/d1;
```

```
u=U*(3*n-n^3)/2;
Tw=(0.323275*rho*U^2)/sqrt(Re);
Re_L=U*L/mu;
Cf=1.292/Re_L^(1/2);
F=Cf*rho*U*U*L/2;
printf('\nThickness of velocity boundary layer at x = 5cm = %f m\nFluid Velocity at y = 0.0225cm = %f m/s\nDrag = %f N\n\n',d1,u,F);
```

Scilab code Exa 5.7 Pressure Change

```
//Example 5.7
//Pressure Change
//Page No. 296
clc;clear;close;

U0=30;            //in m/s
h=80;              //in mm
d1=1;              //in mm
rho=1.23;          //in kg/m^3
dp=(rho*U0*U0/2)*(((h^2)/(h-2*d1)^2)^2-1);
printf('\nPressure Change = %f N/m^2\n\n',dp);
```

Scilab code Exa 5.9 Mass Flow Rate

```
//Example 5.9
//Mass Flow Rate
//Page No. 298
clc;clear;close;

L=1;             //in m
b=0.3;              //in m
U=30;                 //in m/s
```

```
d1=0.0024;          //in m
rho=1.23;           //in kg/m^3
m_ab=rho*U*b*d1/2;
Rx=-1*rho*U*U*b*d1/6;
Rx=-1*Rx;
printf('\nMass flow rate across surface ab = %f kg/s
   \nThe force required to hold the plate in
   position is = %f N\n\n',m_ab,Rx);
printf('\n\n\nNote: Computational errors in book');
```

Scilab code Exa 5.10 Mass flow within boundary layer

```
//Example 5.10
//Mass flow within boundary layer
//Page No. 300
clc;clear;close;

U=2;            //in m/s
x1=0.1;         //in m
x2=0.3;        //in m
rho=1.17;           //in kg/m^3
nu=1.85*10^-5;            //in kg/ms
Re_x1=(rho*U*x1)/nu;
Re_x2=(rho*U*x2)/nu;
d1=4.64*x1/sqrt(Re_x1);
d2=4.64*x2/sqrt(Re_x2);
m=5*rho*U*(d2-d1)/8;
printf('\nMass flow rate = %f kg/s\n\n\n',m);
```

Chapter 7

Turbulent Flow

Scilab code Exa 7.2 Power required

```
//Example 7.2
//Power required
//Page No. 429
clc;clear;close;

Q=1;            //in m^3/s
D_i=0.5;             //in m
rho=1000;            //kg/m^3
nu=1.02*10^-6;            //in m^2/s
g=9.81;              //in m/s^2
U_av=Q/(%pi*D_i^2/4);
Re=U_av*D_i/nu;
f=0.01;
Fric_loss=f*U_av^2/(D_i*2*g);
P=Fric_loss*rho*g*Q;
printf('Power required = %f kW/km',P)
```

Scilab code Exa 7.3 Friction Factor

```
//Example 7.3
//Friction Factor
//Page No. 430
clc;clear;close;

D=60;       // in mm
Ep=1.2;         //in mm
Re=10^8;
Factor=Ep/(D/2);

f=1/(1.74-2*log10(Factor))^2;
Er=1/sqrt(f)-(1.74-2*log10(Factor+18.7/(Re*sqrt(f)))
    );
printf('f = %f \nEr = %f',f,Er);
f1=[0.0485,0.049,0.0475];
for i=1:3
    Er=1/sqrt(f1(i))-(1.74-2*log10(Factor+18.7/(Re*
        sqrt(f1(i)))));
    printf('\n\nf = %f \nEr = %f',f1(i),Er);
end
printf('\n\nSince minimum error value is shown by f
    =0.048605, that is taken to be final answer\nNote
    : Computational error in book')
```

Scilab code Exa 7.4 Developed flow of water

```
//Example 7.4
//Developed flow of water
//Page No. 431
clc;clear;close;

D=6;            //in mm
D=D/100;        //conversion to m
R=D/2;
Q=5*10^-3;      //conversion to m^3/s
```

```
L=10;               //in m
n=7;               //no unit
rho=1000;          //in kg/m^3
nu=1.02*10^-6;     //in m^2/s
U_av=Q/(%pi*D^2/4);
Re=U_av*D/nu;
f=0.3164/Re^(1/4);
Pg=(f*rho*U_av^2)/(2*D);              //Pressure Gradient
Pd=Pg*L;                              //Pressure Drop over 10m
Tw=Pg*R/2;
u_s=U_av*(n+1)*(2*n+1)/(2*n^2);
ds=(Tw*R^(1/7)/(nu*u_s*1000))^(-7/6);       //Thickness of laminar sublayer
printf('\nFriction Factor = %f \nPressure Drop over 10m = %f N/m^2\nThickness of laminar sublayer = %f m',f,Pd,ds);
printf('\n\n\nNote: Slight computational errors in book')
```

Scilab code Exa 7.5 Drag force

```
//Example 7.5
//Drag force
//Page No. 433
clc;clear;close;

U=3;             //in m/s
b=1;              //in m
L=1;              //in m
Re_x=5*10^5;         //no unit
rho=1025;          //in kg/m^3
nu=1.044*10^-6;          //in m^2/s
Re_l=U*L/nu;         //reynolds number on the basis
```

```
   of keel length

//assuming turbulent boundary-layer
Cf=0.074/Re_l^(1/5);
Tw=rho*U^2*Cf/2;
D1=Tw*b*L;
Df=2*D1;
printf('\nTotal Drag Force on the keel (assuming
   turbulent boundary-layer)= %f N',Df);

//taking into account the growth of laminar growtn
   boundary
x_tr=Re_x*nu/U;
d_tr=x_tr*5/sqrt(Re_x);
Cf_lam=1.328/sqrt(Re_x);
D1=rho*U^2*b*(Cf*L-Cf*x_tr+Cf_lam*x_tr)/2;
Df=2*D1;
printf('\nTotal Drag Force on the keel (taking into
   account the growth of laminar growtn boundary)=
   %f N\n\n\nNote: Computational Error in the book',
   Df);
```

Scilab code Exa 7.6 Moody design

```
//Example 7.6
//Moody design
//Page No. 435
clc;clear;close;

U=3;            //in m/s
b=1;             //in m
L=800;          //in m
Re_x=10^6;           //no unit
rho=1000;             //in kg/m^3
nu=1.02*10^-6;        //in m^2/s
```

```
ut_ep_v=100;           //no unit

//calculation via trial and error cannot be shown
    here
x=268;       // = R/e_p
u_t=U/(2.5*log(x)+8.5);
e_p=ut_ep_v*nu/u_t;
R=x*e_p;
D=2*R;
f=0.023;         //no unit
yl=5*nu/u_t;
yb=13*yl;
yl=yl*10^3;     //conversion to mm
yb=yb*10^3;     //conversion to mm
P=%pi*rho*nu*L*u_t^2*Re_x;
printf('\nDiameter = %g m\nLaminar Sub-Layer
    Thickness = %g mm\nBuffer Layer Thickness = %g mm
    \nPower required = %g W',D,yl,yb,P);
```

Chapter 9

Experimental Techniques

Scilab code Exa 9.8 Students t distribution

```
//Example 9.8
//Student's t-distribution
//Page No. 553
clc;clear;close;

n=9;
x_avg=17;
sigma=4;
t=2.31;          //from table A2 in book
printf('\nx = %i +- %.2f\nx_avg = %i +- %.2f',x_avg,
   t*sigma,x_avg,t*sigma/sqrt(n));
```

Scilab code Exa 9.9 Students t distribution

```
//Example 9.9
//Student's t-distribution
//Page No. 553
clc;clear;close;
```

```
n=4;
t=3.18;          //from table A2 in book
x=[65.3,68.2,67.7,66.4];
x_avg=(x(1)+x(2)+x(3)+x(4))/4;
sigma=1.308;
printf('\nx = %.1f +- %.2f\nx_avg = %.1f +- %.3f',
    x_avg,t*sigma,x_avg,t*sigma/sqrt(n));
```