Scilab Textbook Companion for Transmission & Distribution Of Electrical Power by P. Jain[1]

Created by
Harpreet Singh
B TECH
Electrical Engineering
VYAS INSTITUTE OF ENGINEERING AND TECHNOLOGY
College Teacher
NA
Cross-Checked by
K. V. P. Pradeep

May 1, 2014

[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: Transmission & Distribution Of Electrical Power

Author: P. Jain

Publisher: Ashirwad Publications, Jaipur

Edition: 1

Year: 2012

ISBN: 978-93-80343-73-0

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 1

Supply System

Scilab code Exa 1.1 Find the copper saving

```
//Find the copper saving
clear;
clc;
//soltion
//given
v1=240;//volt//initial voltage of the system
v2=500;//volt //final voltage of the system
printf("Volume at 240 volts (vol1) = (4*P^2*d*l)/(%d
    *W)\n",v1^2/4);
printf("Volume at 500 volts (vol2) = (4*P^2*d*l)/(%d
    *W)\n",v2^2/4);
printf("Percentage saving in copper = ((vol1-vol2)
    *100)/vol1\n");
s = (((1/v1^2)-(1/v2^2))/(1/v1^2))*100;
printf("The percentage saving of the copper is , %.2f
     percent",s);
```

Scilab code Exa 1.4 Calculate volume of conductor required in 1 phase 2 wire and 3 phase 3 wire system

```
//Calculate volume of conductor required in 1 phase 2 wire and 3 phase 3 wire system
clear;
clc;
//solution
//given
pf=0.8;//power factor
pMVA=(2.5*10^6);//volt ampere
v=(33*10^3);//volts
l=50*10^3;//m//length of the line
p=pMVA*pf;//watts//power trasmitted = power in MVA*p.f.
w=0.2*p;//watts//line losses = 20% of power transmitted
d=2.85/10^8;//ohm meter//resistivity of aluminium
printf(" 1 phase 2 wire system\n");
i1=pMVA/v;
a1=(2*i1^2*d*l)/w;
printf(" Load current in 1 phase 2 wire system= %f ampere\n",i1);
printf(" Cross sectional area of 1 phase 2 wire system= %f m^2\n",a1);
vol1=2*a1*l;
printf(" Volume of aluminium conductor required in 1 phase 2 wire system = %f meter cube \n\n",vol1);
printf(" 3 phase 3 wire system\n");
i2=pMVA/(3^0.5*v);
a2=(3*i2^2*d*l)/w;
printf(" Load current in 3 phase 3 wire system= %f ampere\n",i2);
printf(" Cross sectional area of 3 phase 3 wire system= %f m^2\n",a2);
vol2=3*a2*l;
printf("Volume of aluminium conductor required in 3 phase 3 wire system = %f meter cube ",vol2);
```

Scilab code Exa 1.5 Calculate DC supply voltage

```
//find the DC supply voltage
clear;
clc;
//soltion
//given
//consider 1 phase AC system
pf=0.8;
v=(33*10^3);//volts
r1=0.15;//ohm//total resistance of the 1 phase line
PD1=0.2;//percentage voltage drop in 1 phase AC system
Vd=PD1*v;//volt//voltage drop in the line
I1=Vd/r1;//ampere//load current
p=v*I1*pf;//watts//power recieved by the consumer
P=p/10^8;
printf("1 phase AC system \n");
printf("Voltage drop= %d volts\n",Vd);
printf("Load current= %d ampere\n",I1);
printf("Power recieved by consumer= %d watts or= %f *10^5 kW \n\n",p,P);
//consider DC 2 wire system
r2=0.1;//ohm//total resistance of the DC 2 wire line
PD2=0.25;//percentage voltage drop in DC 2 wire system
printf("DC 2 wire system\n");
printf("Load current in DC system= %f/V \n",p);
printf("Voltage drop= Load curret*line resistance= I2*R2= (%d/V)*%f \n", p, r2);
printf("Given voltage drop is 25 percentage of max voltage= .25*V \n ");
V=sqrt((p*r2)/PD2);
printf("Equating above equation we get V= %f KV",V
```

```
/1000);
```

Chapter 2

Distribution System

Scilab code Exa 2.1 Calculate the most economical cross sectional area

```
//Calculate the most economical cross sectional area
clear;
clc;
//soltion
//given
id=0.15;//interest & depreciation charges
i=260;//ampere//max current
d=0.173;//ohm//resistance of conductor
cst=.03;//rs// cost of energy per unit
t=(365*24)/2;//time of energy loss
printf("Annual cost of 2 core feeder cable is Rs(90a+10)per meter\n");
P3=(2*i^2*d*t*cst)/1000//kWh//annual cost of energy loss
printf("Energy loss per annum= P3/a= %f/a \n",P3);
P2=90*1000*id;//energy lost per annum
printf("Capital cost= P2*a= %d*a \n", P2);
a=sqrt(P3/P2);
printf("Economic cross section of conductor is=     (P3/P2)= %f square cm",a);
```

Scilab code Exa 2.2 Calculate the most economical current density

```
//Calculate the most economical current density
clear;
clc;
//soltion
//given
id=0.1;//interest & depreciation charges
d=1.78*10^-8;//ohm m//resistivity
R=(d*1000)/10^-4;//ohm//resistance of conductor
cst=.50;//rs// cost of energy per unit
t=(365*24);//time of energy loss
lf=.7;//load factor of losses
printf("Annual cost of cable is Rs(2800a+1300)per km
   \n\n");
printf("Resistance of each conductor= %f/a \n", R);
P3=(R*t*cst*lf)/1000;//*I^2//kWh//annual cost of
   energy loss
printf("Annual cost of energy loss= P3/a= (%f*I^2)/a
    \n", P3);
P2=2800*id;//energy lost per annum
printf("Annual charge on account of intrest and
   depreciation on variable cost of line= P2*a= %d*a
   \n", P2);
J=sqrt(P2/P3);//current density I/a
printf("Economic current density of conductor is %f
   A/cm square",J);
```

Scilab code Exa 2.3 Calculate the most economical current density and diameter of conductor

```
//Calculate the most economical current density and
    diameter of conductor
clear;
clc;
//soltion
//given
id=0.1;//interest & depreciation charges
cst=.02;//rs// cost of energy per unit
d=0.173;//ohm//resistance of conductor
pf=.8;//lagging
P=1500*10^3;//Watts//load
V=11000;//volts//supply voltage
t=200*8;//hours
printf("annual cost of 3 core feeder cable is Rs
    (8000 + 20000 a)per km\n");
printf("Resistance of each conductor= %.3f/a \n", d)
    ;
i=P/(sqrt(3)*V*pf);//ampere
printf("Current in each conductor= %.3f A\n", i);
P2=20000*id;//energy lost per annum
printf("Capital cost= P2*a= %d*a \n", P2);
P3=(3*i^2*d*t*cst)/1000;//kWh//annual cost of energy
     loss
printf("Energy loss per annum= P3/a= %f/a \n",P3);
a=sqrt(P3/P2);
printf("Economic cross section of conductor is=     (
    P3/P2)= %f square cm \n",a);
printf("Diameter of conductor= %.1f cm \n", sqrt(4*a
    /%pi));
printf("Current density= %f A/cm square", i/a);
```

Scilab code Exa 2.4 Calculate the most economical cross sectional area

```
//Calculate the most economical cross sectional area
clear;
```

```
clc;
//soltion
//given
id=0.1;//interest & depreciation charges
pf=.8;//lagging
P=10^6;//Watts//load
V=11000;//volts//supply voltage
cst=.15;//rs// cost of energy per unit
d=1.75*10^-6;//ohm cm//specific resistance
l=1000//m//length of the cable
t=3000;//hours
printf("Annual cost of 2 core feeder cable is Rs(30 + 500a)per meter\n");
R=(d*1000*100);//ohm//resistance of conductor
printf("Resistance of each conductor= %f/a \n", R);
i=P/(V*pf);//ampere
printf("Current in each conductor= %f A\n", i);
P2=500*10^3*id;//energy lost per annum
printf("Capital cost= P2*a= %d*a \n", P2);
P3=(2*i^2*R*t*cst)/1000;//kWh//annual cost of energy loss
printf("Energy loss per annum= P3/a= %f/a \n",P3);
a=sqrt(P3/P2);
printf("Economic cross section of conductor is=     (P3/P2)= %f square cm \n",a);
printf("Diameter of conductor= %f cm \n", sqrt(4*a/%pi));
```

Scilab code Exa 2.5 Calculate the most economical cross sectional area

```

//Calculate the most economical cross sectional area
clear;
clc;
//soltion
```

```
//given
id=0.1;//interest & depreciation charges
pf=.85;//lagging
Pm=10^3;//Watts//Max Demand
Pt=5*10^6//kWh//Toatal energy consumption
V=11000;//volts//supply voltage
cst=.05;//rs// cost of energy per unit
d=1.72*10^-6;//ohm cm//specific resistance
t=(365*24);//time of energy loss
printf("Annual cost of cable is Rs(80000a + 20000) per km\n");
lf=Pt/(Pm*t)//Annual load factor
printf("Annual load factor= %f\n", lf);
llf=.25*lf+.75*lf^2;//Loss load factor
printf("Loss load factor= %f\n",llf);
i=Pm*1000/(sqrt(3)*V*pf);//ampere
printf("Current in each conductor= %.1f A\n", i);
P2=80000*id;//energy lost per annum
printf("Capital cost= P2*a= %d*a*l \n", P2);
R=d*100*1000;//ohm
P3=(3*i^2*R*t*cst*llf)/1000;//kWh//annual cost of energy loss
printf("Energy loss per annum= (P3*l)/a= (%f*l)/a \n",P3);
a=sqrt(P3/P2);
printf("Economic cross section of conductor is=     (P3/P2)= %f square cm \n",a);
//THERE IS TYPOGRAPHICAL ERROR IN THE ANS IN BOOK IT IS 0.2404 cm^2
```

Scilab code Exa 2.6 Calculate the most economical cross sectional area

```
//Calculate the most economical cross sectional area
clear;
clc;
```

```
//soltion
//given
id=0.1;//interest & depreciation charges
V=20000;//volts//supply voltage
d=1.72*10^-6;//ohm cm//specific resistance
cst=.6;//rs// cost of energy per unit
p1=1500//kilowatts
t1=8//hours
pf1=.8//power factor
p2=1000//kilowatts
t2=10//hours
pf2=.9//power factor
p3=100//kilowatts
t3=6//hours
pf3=1//power factor
t=365//no. of days
i1=p1*1000/(sqrt(3)*V*pf1);//ampere//current at time t1
i2=p2*1000/(sqrt(3)*V*pf2);//ampere//current at time t2
i3=p3*1000/(sqrt(3)*V*pf3);//ampere//current at time t3
R=d*100*1000;//ohm
P2=8000*id;//Loss load factor
printf("Annual cost of cable is Rs(80000a + 20000) per km\n");
printf("Capital cost= P2*a= %d*a*l \n", P2);
P3=(3*((i1^2*t1)+(i2^2*t2)+(i3^2*t3))*R*t*cst)/1000; //kWh//annual cost of energy loss
printf("Energy loss per annum= (P3*l)/a= (%f*l)/a \n",P3);
a=sqrt(P3/P2);
printf("Economic cross section of conductor is=    (P3/P2)= %f square cm \n",a);
```

Chapter 3

Mechanical Features of Overhead Line

Scilab code Exa 3.1 calculate the weight of the conductor required

```
//calculate the weight of the conductor required
clear;
clc;
//soltion
//given
p=30*10^6;//watts// power to be transmitted
v=132*10^3;//volts//Line voltage
l=120*10^3;//m//length of 3 phase 3 wire line
n=0.9;//efficieny of the transmission line
pf=.8;//power factor
d1=1.78*10^-8;//ohm m//resistivity of copper
d2=2.6*10^-8;//ohm m//resistivity of aluminuim
D1=8.9*10^3;//(kg/m^3)//specific gravity of the
    copper
D2=2*10^3;//(kg/m^3)//specific gravity of the
    aluminium
printf("Weight of the conductor required \n\n");
printf("W=(3*d*l^2*P*D)/((1-n)*V^2*pf^2) kg\n\n\n");
W1=(3*d1*l^2*p*D1)/((1-n)*v^2*pf^2);
```

```
printf("Weight of coppper required= %d kg\n\n",round
    (W1/1000)*1000);
W2=(3*d2*l^2*p*D2)/((1-n)*v^2*pf^2);
printf("Weight of aluminuim required= %d kg\n\n\n",
    round(W2/100)*100);
```

Scilab code Exa 3.2 Calculate the max sag

```
//Calculate the max sag
clear;
clc;
//soltion
//given
W=.6;//kg/m//Line conductor wieght
L=300;//meter//span of the line
T=1200;//kg//max allowable tension
printf("Max sag= (W*L^2)/(8*T)\n");
sag= (W*L^2)/(8*T);
printf("Sag= %.3f m",sag);
```

Scilab code Exa 3.3 Calculate the hieght above ground at which conductor should be supported

```
//Calculate the hieght above ground at which
    conductor should be supported
clear;
clc;
//soltion
//given
W=680;//kg/km//Line conductor weight
L=240;//meter//span of the line
U=3200;//kg//Ultimate strength
sf=2;//safety factor
```

```
T=U/sf;//kg//max allowable tension
gc=8;//m//ground clearance
w=W/1000;//kg/m//Weight of conductor in meter
printf("Max sag= (W*L^2)/(8*T)\n");
sag= (w*L^2)/(8*T);
printf("Sag= %.2f m\n",sag);
H=gc+sag;
printf("Height above which conductor should be supported\n= ground clearance+ sag= %.2f m",H);
```

Scilab code Exa 3.4 Calculate horizontal component of tension and max sag

```
//Calculate horizontal component of tension and max sag
clear;
clc;
//soltion
//given
W=750;//kg/km//Line conductor weight
L=300;//meter//span of the line
T=3400;//kg//max allowable tension
w=W/1000;//kg/m//Weight of conductor in meter
printf("Max sag= (W*L^2)/(8*Th)\n");
x= (w*L^2)/(8);
printf("Sag= %.1f/Th\n\n",x);
printf("Max tension= Th + wS\n");
Th=(T+sqrt(T^2+4*w*x))/2;
printf("Horizontal component of tension (Th)= %.3f kg\n",Th);
sag= (w*L^2)/(8*Th);
printf("Sag= %.3f m\n",sag);
y=sag/2;
z=sqrt((2*Th*y)/w);
printf("Point at which sag will be half= %.3f m\n",
```

```
    z);
//THERE IS TYPOGRAPHICAL ERROR IN BOOK DUE TO THAT
    THERE IS A VARIATION
//IN BOOK Th=3448.191 kg
//MAX SAG=2.446 m
//Point at which sag will be half= 106.045 m
```

Scilab code Exa 3.5 Calculate the max sag in still air and wind pressure

```
//Calculate the max sag in still air and wind
    pressure
clear;
clc;
//soltion
//given
Wc=1.13;//kg/m//Line conductor weight
P=33.7//kg/m^2//Wind pressure
L=180;//meter//span of the line
fu=4220;//kg//Ultimate stress
sf=5;//safety factor
f=fu/sf;//kg//working stress
D=1.27;//cm//dia of copper
r=1.25;//cm//Radial thickness of ice
a=(%pi*D^2)/4;//cm^2//area of cross section
printf("Area of cross section= %3f cm^2\n",a);
T=f*a;//kg//max allowable tension
printf("Working tension= %.2f kg\n",T);
sag1= (Wc*L^2)/(8*T);//sag in still air
printf("Sag in sill air= %.2f m\n",sag1);
Wi=2890.3*r*(D+r)*10^-4;
printf("Weight of ice coating= %.2fkg\n",Wi);
Ww=P*(D+2*r)*10^-2;
printf("Wind force= %.5fkg\n",Ww);
Wr=sqrt((Wc+Wi)^2+Ww^2);
sag2= (Wr*L^2)/(8*T);//sag in wind + ice
```

```
printf("Max Sag= %.3f m\n",sag2);
```

Scilab code Exa 3.6 Calculate the max sag

```
//Calculate the max sag
clear;
clc;
Wc=.85;//kg/m//Line conductor wieght
L=275;//meter//span of the line
U=8000;//kg//Ultimate strength
sf=2;//safety factor
P=39;//kg/m^2//Wind pressure
T=U/sf;//kg//max allowable tension
D=19.5;//mm//dia of copper
r=13;//cm//Radial thickness of ice
Wi=910*%pi*r*(D+r)*10^-6;
Ww=P*(D+2*r)*10^-3;//Wind force/m lenght
Wr=sqrt((Wc+Wi)^2+Ww^2);//resultant sag
sag= (Wr*L^2)/(8*T);//sag in wind + ice
printf("Max Sag= %.3f m\n",sag);
```

Scilab code Exa 3.7 Calculate the vertical sag

```
//Calculate the vertical sag
clear;
clc;
//soltion
//given
W=1170;//kg/km//Line conductor wieght
P=122;//kg/m^2//Wind pressure
L=200;//meter//span of the line
A=1.29;//cm^2//cross sectional area
U=4218*A;//kg//Breaking strength
```

```
sf=5;//safety factor
T=U/sf;//kg//max allowable tension
Wc=W/1000;//kg/m//Weight of conductor in meter
D=sqrt((4*A)/%pi);//cm//diameter of the conductor
printf("Diameter of the conductor= %.2f cm\n",D);
Ww=P*(D)*10^-2;//Wind force/m lenght
printf("Wind force= %.2fkg\n",Ww);
Wr=sqrt(Wc^2+Ww^2);//resultant weight
printf("Resultant sag= %.2fkg\n",Wr);
sag= (Wr*L^2)/(8*T);//m//Slant sag
printf("Slant Sag= %.2f m\n",sag);
Th=atand(Ww/Wc);//degree//angle between slant sag and vertical sag
Vsag=sag*cosd(Th);//m//Vertical sag
printf("Vertical sag= %.3fm",Vsag);
```

Scilab code Exa 3.8 Calculate the minimum clearance of conductor and water

```
//Calculate the minimum clearance of conductor and water
clear;
clc;
//soltion
//given
W=1.5;//kg/m//Line conductor wieght
L=500;//meter//span of the line
T=1600;//kg//max allowable tension
T1=30;//m//height of the tower 1
T2=90;//m//height of the tower 2
h=T2-T1;//m//difference in the between support
printf("Distance of support T1 from O(Lowest point) be x1\n");
printf("Distance of support T2 from O(Lowest point) be x2\n");
```

```
printf("x1+x2= %dm\n",L);
dif=((h*2*T)/(W*L));//x2-x1
printf("x2-x1= %dm\n",dif);
x2=(L+dif)/2;//m
x1=L-x2;//m
printf("x1= %dm, x2= %dm\n",x1,x2);
sag= ((W*x1^2)/(2*T));//m
printf("Sag(From tower 1)= %d m\n",round(sag));
C=T1-sag;//Clearance
printf("Clearance of the lowest point from water level= %dm\n",C);
```

Scilab code Exa 3.9 Calculate sag from taller of the two supports

```
//Calculate sag from taller of the two supports
clear;
clc;
//soltion
//given
Wc=1.925;//kg/m//Line conductor wieght
L=600;//meter//span of the line
h=15//m//T1-T2
Wi=1//kg//Wieght of the ice
Wr=Wi+Wc;//resultant weight
A=2.2//cm^2
U=8000*A;//kg//Breaking strength
sf=5;//safety factor
T=U/sf;//kg//max allowable tension
printf("x1+x2= %dm\n",L);
dif=((h*2*T)/(Wr*L));//x2-x1
printf("x2-x1= %dm\n",dif);
x2=(L+dif)/2;//m
x1=L-x2;//m
printf("x1= %dm, x2= %dm\n",round(x1),round(x2));
sag= ((Wr*(round(x2))^2)/(2*T));//m
```

```
printf("Sag(from taller of the two supports)= %.3f m\n",sag);
```

Scilab code Exa 3.10 find the clearance of conductor from ground

```
//find the clearance of conductor from ground
clear;
clc;
//soltion
//given
W=1;//kg/m//Line conductor wieght
L=300;//meter//span of the line
T=1500;//kg//max allowable tension
T1=22-2;//m//effective height of the towers
g=1/20;//sin    //gradient
h=L*g//m//vertical distance between two towers
printf("x1+x2   %dm\n",L);
dif=((h*2*T)/(W*L));//x2-x1
printf("x2-x1= %dm\n",dif);
x2=(L+dif)/2;//m
x1=L-x2;//m
printf("x1= %dm, x2= %dm\n",round(x1),round(x2));
sag= ((W*x2^2)/(2*T));//m
printf("Sag= %.3f m\n",sag);
T2=T1+h;//m//hieght of the second tower
gf=x1*tand(asind(1/20));//m//elevation of the groundat max sag
OG=T2-sag-gf;//m//ground clearance
printf("Clearance of the lowest point O from ground is %.2fm",OG);
//SINCE THERE IS NO REFRENCE OF WATERLEVEL IN THE QUESTION THEREFORE THE EXTRA SOLUTION IS AN TYPOGRAPHICAL ERROR
```

Scilab code Exa 3.11 Find stringing tension in the conductor

```
//Find stringing tension in the conductor
clear;
clc;
//soltion
//given
W=0.7;//kg/m//Line conductor wieght
L=250;//meter//span of the line
T1=25;//m//height of the tower 1
T2=75;//m//height of the tower 2
h=T2-T1;//m//difference in the between support
Tm=45;//m//hieght of midway between the towers
hm=Tm-T1;//m//midway point between the two towers
Lm=L/2;//m//half of the span
printf("We know that \nh=(W*L*(L-2*x))/(2*T)\n");
printf("For the two towers\n%d=(%.1f*%d(%d-2*x))/(2*T)\n",h,W,L,L);
printf("For the mid point \n%d=(%.1f*%d(%d-2*x))/(2*T)\n",hm,W,Lm,Lm);
x=-(2*L)+(2.5*Lm);
printf("By above equation x= %d m\n",x);
T=(W*L*(L-2*x))/(2*h);
printf("Stringing Tension(T)=%.2f kg",T)
```

Scilab code Exa 3.12 find the clearance of conductor from water level at mid point

```
//find the clearance of conductor from water level at mid point
clear;
clc;
```

```
//soltion
//given
W=.844;//kg/m//Line conductor wieght
L=300;//meter//span of the line
T=1800;//kg//max allowable tension
T1=40;//m//height of the tower 1
T2=80;//m//height of the tower 2
h=T2-T1;//m//difference in the between support
x=L/2-(T*h)/(W*L);
printf("Distance between midpoint and lowest point=%.2fm\n",(L/2)-x);
Smid=(W*(L/2-x)^2)/(2*T);
printf("Height between midpoint and lowest point= %.3fm\n",Smid);
S2=(W*(L-x)^2)/(2*T);
printf("Height between taller tower and lowest point = %.3fm\n",S2);
C=T2-(S2-Smid);
printf("Clearance of conductor from water level at mid point= %.3fm",C)
```

Scilab code Exa 3.13 find the clearance of conductor from ground 1 At its lowest elevation 2 the min clearance of the line

```
//find the clearance of conductor from ground i)At its lowest elevation ii)the min clearance of the line
clear;
clc;
//soltion
//given
W=.8;//kg/m//Line conductor wieght
L=300;//meter//span of the line
T=1500;//kg//max allowable tension
T1=30;//m//height of the towers
```

```
g=1/20;//tan   //ground slope
h=L*g//m//vertical distance between two towers
T2=T1+h;//m//height of the tower along the slope
x1=L/2-(T*h)/(W*L);
printf("Distance between tower on ground and sag=x1=%.2fm\n",x1);
S1=(W*x1^2)/(2*T);
printf("Sag for tower on ground(S1)= %.5fm\n",S1);
S2=(W*(L-x1)^2)/(2*T);
printf("Sag for tower on hill(S2)= %.5fm\n",S2);
C=T1-S1-x1*g;
printf("Clearance of conductor from lowest elevation = %.5fm\n",C);
x=poly(0, 'x');
C1= poly([C -g W/(2*T)],'x','c');
d=derivat(C1);
xa=roots(d);
Ca=C-g*xa+W/(2*T)*xa^2;
printf("Minimum clearance from ground= %dm",Ca);
```

Scilab code Exa 3.14 Determine Sag and Tension under erection conditions

```
//Determine Sag & Tension under erection conditions
clear;
clc;
//soltion
//given
W=.9;//kg/m//Line conductor wieght
L=300;//meter//span of the line
a=2.40*10^-4//m^2//area
D=19.5//mm//diameter
U=8000;//kg//Ultimate strength
sf=2;//safety factor
P=38.5;//kg/m^2//Wind pressure
```

```
T1=U/sf;//kg//max allowable tension
E=9320*10^6;//kg/m^2//Young's Modulus
alp=18.44*10^-6;//1/ C //Linear expansion
t1=5// C //temperature under normal condition
t2=35// C //temperature under worst condition
dt=t2-t1;// C //difference in temperature
f1=T1/a;
Ww=P*(D)*10^-3;//weight due to wind
printf("Wind force= %.2fkg\n",Ww);
Wr=sqrt(W^2+Ww^2);//resultant weight
C1=W^2*L^2*E/(24*a^2);
C2=-f1+Wr^2*L^2*E/(24*f1^2*a^2)+dt*alp*E;
p=poly([-C1 0 C2 1], 'f2', 'c');
r=roots(p);
f2= 11951292;//accepted value of f2
sag=(W*L^2)/(8*f2*a);
printf("Sag at erection= %.3fm",sag);
//The book has used in correct value of f2 and in it
    the sag is 2.121m
```

Chapter 4

Transmission Line Parameters

Scilab code Exa 4.1 Find the loop inductance and reactance

```
//Find the loop inductance and reactance
clear;
clc;
//soltion
//given
r=(1.213*10^-2)/2;//m//radius of the conductor
d=1.25;//m//spacing
f=50;//Hz//freq
re=r*exp(-1/4);
L=4*10^-7*log(d/re);
Lkm=L*1000;
printf("Inductance per km(L)=%.2f*10^-4 H/Km\n",Lkm
   *10^4);
X=2*%pi*f*Lkm;
printf("Reactance(X)= %.1f ohm/km",X);
```

Scilab code Exa 4.2 Find the loop inductance

```
//Find the loop inductance
clear;
clc;
//soltion
//given
r=(1*10^-2)/2;//m//radius of the conductor
d=2;//m//spacing
u=50//relative permeability of steel and copper
L=(1+4*log(d/r))*10^-7*1000;
LmH=L*1000;
printf("Inductance per km(L) copper conductor=%.3f
   mH\n",LmH);
Lr=(u+4*log(d/r))*10^-7*1000;
printf("Inductance per km(L) steel conductor=%.3f mH
   \n",Lr*1000);
```

Scilab code Exa 4.3 Calculate GMR pf ACSR conductor

```
//Calculate GMR pf ACSR conductor
clear;
clc;
//soltion
//given
r=3;//mm//radius of the conductor
re=r*exp(-1/4);
d11=re;
d12=2*r//=d17=d16;
d14=4*r;
d13=sqrt(d14^2-d12^2);//=d15
Ds1=(d11*d12*d13*d14*d13*d12*d12);
Ds1_=Ds1/(r^7);
printf("Ds1= (%f)^(1/7)*r\n",Ds1_);
d71=2*r;//=d72=d73=d74=d75=d76
Ds7=(d71^6*re);
Ds7_=Ds7/(r^7);
```

```
printf("Ds7= (%f)^(1/7)*r\n",Ds7_);
Ds=(Ds1^6*Ds7)^(1/49);
printf("GMR= %.4fmm",Ds);
```

Scilab code Exa 4.4 Find the total inductance of the line

```
//Find the total inductance of the line
clear;
clc;
//soltion
//given
r=1.4;//cm//radius of the conductor
re=r*exp(-1/4);
d12=20;//cm//spacing b/w 1&2
d11_=20+120;//cm//spacing b/w 1&1'
d12_=20+120+20;//cm//spacing b/w 1&2'
d21_=120;//cm//spacing b/w 2&1'
d22_=20+120;//cm//spacing b/w 2&2'
Dm=(d11_*d12_*d21_*d22_)^(1/4);
printf("Mutual GMD= %.2fcm\n",Dm);
Ds=floor((re*d12*re*d12)^(1/4)*100)/100;
printf("Self GMD= %.2fcm\n",Ds);
L=0.4*log(Dm/Ds);
printf("Loop Inductance of line= %.5f mH/km",L);
```

Scilab code Exa 4.5 Find the loop inductance

```
//Find the loop inductance
clear;
clc;
//soltion
//given
r=1/2;//cm//radius of the conductor
```

```
re=r*exp(-1/4);
d12=200;//cm//spacing b/w 1&2
d11_=300;//cm//spacing b/w 1&1'
d12_=sqrt((300)^2+(200)^2);//cm//spacing b/w 1&2'
d21_=d12_;//cm//spacing b/w 2&1'
d22_=300;//cm//spacing b/w 2&2'
Dm=(d11_*d12_*d21_*d22_)^(1/4);
printf("Mutual GMD= %.3fcm\n",Dm);
Ds=(re*d12*re*d12)^(1/4);
printf("Self GMD= %.3fcm\n",Ds);
L=0.4*log(Dm/Ds);
printf("Loop Inductance of line= %.3fmH/km\n",L);
```

Scilab code Exa 4.6 Find the inductance per phase of 30 km line

```
//Find the inductance per phase of 30 km line
clear;
clc;
//soltion
//given
r=(15)/2;//mm//radius of the conductor
re=r*exp(-1/4);
d=1.5*1000;//mm//spacing
L=0.2*log(d/re);
printf("Loop Inductance of line= %.2f mH/km\n",L);
L1=L*30/1000;
printf("Inductance per phase of 30 km long line= %.4
   f H",L1);
```

Scilab code Exa 4.7 Find the inductance of a 3 phase line situated at cornes of a triangle

```
//Find the inductance of a 3 phase line(triangle)
```

```
clear;
clc;
//soltion
//given
r=1;//cm//radius of the conductor
re=r*exp(-1/4);
d1=600;//cm//spacing of the triangular shaped system
d2=700;//cm//spacing of the triangular shaped system
d3=800;//cm//spacing of the triangular shaped system
L=0.2*log(((d1*d2*d3)^(1/3))/re);
printf("Loop Inductance of line= %.4f mH/km\n",L);
```

Scilab code Exa 4.8 Find the inductance of a 3 phase line arranged in horizontal plane

```
//Find the inductance of a 3 phase line(plane)
clear;
clc;
//soltion
//given
r=1;//cm//radius of the conductor
re=r*exp(-1/4);
d=300;//cm//spacing b/w conductors
C1=0.2*[log(d/re)+0.5*log(2)];
C2=0.2*((sqrt(3))/2)*log(2);
La=complex(C1,-C2);
Lb=0.2*log(d/re);
Lc=complex(C1,C2);
printf("La= (%.2f %.2fj)mH\n",real(La),imag(La));
printf("Lb= %.4fmH\n",Lb);
printf("Lc= (%.2f +%.2fj)mH\n",real(Lc),imag(Lc));
```

Scilab code Exa 4.9 Find the loop inductance per phase

```
//Find the loop inductance per phase
clear;
clc;
//soltion
//given
r=5;//mm//radius of the conductor
re=r*exp(-1/4);
d=3500;//mm//spacing
L=2*10^(-7)*log(d/re);
L_=L*10^6;
printf("Inductance per km(L)=%.4f*10^-6 H\n",L_);
printf("Lav=2*10^-7{log(dp/r)+1/3*log(2)}\n");
printf("Lav= L\n");
Z=(L/(2*10^-7)-1/3*log(2));
dp=re*exp(Z);
dp_=dp/1000;
printf("After soving above equation\n");
printf("Spacing between the conductors in the plane(dp)= %.3fm",dp_);
```

Scilab code Exa 4.10 Find the loop inductance per phase

```
//Find the loop inductance per phase
clear;
clc;
//soltion
//given
r=20;//mm//radius of the conductor
re=r*exp(-1/4);
d=7000;//mm//spacing
L=0.1*log((sqrt(3))*d/(2*re));
printf("Inductance per km(L)=%.4f mH\n",L);
```

Scilab code Exa 4.11 Find the inductance of an ASCR 3 phase line

```
//Find the inductance of an ASCR 3 phase line
clear;
clc;
//soltion
//given
r=5/2;//mm//radius of the conductor
re=r*2.177*10^-3;//m
dx=6;//m//spacing in X direction
dy=8;//m//spacing in Y direction
daa_=sqrt(dx^2+(2*dy)^2);
dbb_=6;
dcc_=daa_;
dab=8;
dab_=sqrt(dx^2+dy^2);
dbc=8;
dbc_=sqrt(dx^2+dy^2);
dca_=6;
dca=16;
Dsa=sqrt(re*daa_);
Dsb=sqrt(re*dbb_);
Dsc=sqrt(re*dcc_);
Ds=(Dsa*Dsb*Dsc)^(1/3);
printf("Self GMD or GMR, Ds= %.4fm\n",Ds);
Dab=sqrt(dab*dab_);
Dbc=sqrt(dbc*dbc_);
Dca=sqrt(dca*dca_);
Dm=(Dab*Dbc*Dca)^(1/3);
printf("GMD, Dm= %.2fm\n",Dm);
L=0.2*log(Dm/Ds);
printf("Inductance of 100 km line (L)=%.4f H\n",L
    *0.1);
L_=0.1*log((2^(1/3))*(dy/re)*((dx^2+dy^2)/(4*dy^2+dx
    ^2))^(1/3));
printf("Inductance(By another method) per phase per
    km(L)=%.4f H\n",L_*.1);
```

Scilab code Exa 4.12 Find inductive reactance of 3 phase bundled conductor

```
//Find inductive reactance of 3 phase bundled
   conductor
clear;
clc;
//soltion
//given
r=1.75*10^-2;//m//radius of the conductor
re=r*exp(-1/4);
d=7;//spacing
S=0.4;//spacing between subconductors
Ds=sqrt(re*S);//GMR
dab=7;
dab_=7.4;
da_b=6.6;
da_b_=7;
Dab=(dab*dab_*da_b*da_b_)^.25;
Dbc=Dab;
dca=14;
dca_=13.6;
dc_a=14.4;
dc_a_=14;
Dca=(dca*dca_*dc_a*dc_a_)^.25;
Dm=(Dab*Dca*Dbc)^(1/3);//GMD
L=0.2*log(Dm/Ds);
printf("Inductance(L)=%.4f mH/km\n",L);
Xl=2*%pi*50*L*10^-3;
printf("Inductive reactance= %.1f   /km\n",Xl);
r1=sqrt(2*((r*10^2)^2));
re1=r1*exp(-1/4);
Dab1=7;
Dbc1=7;
```

```
Dca1=14;
Dm1=(Dab1*Dbc1*Dca1)^(1/3);//GMD of single conductor line
L1=0.2*log(Dm1/(re1*10^-2));
printf("Inductance(L)=%.3f mH/km\n",L1);
Xl1=2*%pi*50*L1*10^-3;
printf("Inductive reactance= %.3f   /km",Xl1);
```

Scilab code Exa 4.13 Find the capacitance of 1 phase line

```
//Find the capacitance of 1 phase line
clear;
clc;
//soltion
//given
r=15/2;//mm//radius of the conductor
d=1500;//mm//spacing
L=30000;//m//length of the line
Eo=8.85*10^-12//permitivity of the air
C=%pi*Eo*L/(log(d/r));
C_=C*10^6;
printf("Capacitance of 30km line= %f  F ",C_);
```

Scilab code Exa 4.14 Find the capacitance of 2 wire 1 phase line

```
//Find the capacitance of 2 wire 1 phase line
clear;
clc;
//soltion
//given
r=0.25;//cm//radius of the conductor
d=150;//cm//spacing
L=50000;//m//length of the line
```

```
h=700//cm//height of conductor above earth
Eo=8.854*10^-12//permitivity of the air
C=%pi*Eo*L/(log(120/(sqrt(1+(d^2/(2*h)^2))*r)));
C_=C*10^6;
printf("Capacitance of 50km line= %.3f F",C_);
```

Scilab code Exa 4.15 Find the capacitance of 3 phase line

```
//Find the capacitance of 3 phase line
clear;
clc;
//soltion
//given
r=1;//cm//radius of the conductor
d=250;//cm//spacing
L=100000;//m//length of the line
Eo=8.854*10^-12//permitivity of the air
C=2*%pi*Eo*L/(log(d/r));
C_=C*10^6;
printf("Capacitance of 100km line= %.4f F",C_);
```

Scilab code Exa 4.16 Find the capacitance of 3 phase 3 wire line

```
//Find the capacitance of 3 phase 3 wire line
clear;
clc;
//soltion
//given
r=0.01;//m//radius of the conductor
d1=3.5;//m//spacing
d2=5;//m//spacing
d3=8;//m//spacing
L=1000;//m//length of the line
```

```
Eo=8.854*10^-12//permitivity of the air
de=(d1*d2*d3)^(1/3)
C=2*%pi*Eo*L/(log(de/r));
C_=C*10^6;
printf("Capacitance of line= %.4f   F",C_);
```

Scilab code Exa 4.17 Find the capacitance and charging current

```
//Find the capacitance and charging current
clear;
clc;
//soltion
//given
f=50;//frequency
Vph=220*1000/sqrt(3);//phase voltage
r=0.01;//m//radius of the conductor
d1=3;//m//spacing
d2=3;//m//spacing
d3=6;//m//spacing
L=1000;//m//length of the line
Eo=8.854*10^-12//permitivity of the air
de=(d1*d2*d3)^(1/3)
C=2*%pi*Eo*L/(log(de/r));
C_=C*10^9;
printf("Capacitance of line= %.4f*10^-12F\n",C_);
Ic=2*%pi*f*C*Vph;
printf("Charging current per phase is= %.3fmA",Ic);
```

Scilab code Exa 4.18 find capacitive reactance to neutral and charging current

```
//find capacitive reactance to neutral and charging
   current
```

```
clear;
clc;
//soltion
//given
r=1.25*10^-2;//m//radius of the conductor
f=50//frequency
Vph=132*1000/sqrt(3);//phase voltage
Eo=8.85*10^-12//permitivity of the air
drr_=sqrt(7^2+(4+4)^2);
dbb_=drr_;
dyy_=9;
Dsr=sqrt(r*drr_);
Dsy=sqrt(r*dyy_);
Dsb=sqrt(r*dbb_);
Ds=(Dsr*Dsy*Dsb)^(1/3);
dry=sqrt(4^2+(4.5-3.5)^2);
dry_=sqrt((9-1)^2+4^2);
Dry=sqrt(dry*dry_);
Dyb=Dry;
Dbr=sqrt(8*7);
Dm=(Dyb*Dbr*Dry)^(1/3);
C=2*%pi*Eo/(log(Dm/Ds));
printf("Capacitance per phase= %.2f*10^-9 F/km\n",C
    *10^12);
Cr=1/(2*%pi*f*C*1000);
printf("Capacitance per phase= %.2f k \n",Cr/1000);
Ic=(2*%pi*f*C*1000)*Vph;
printf("Charging current= %.4f A/km",Ic);
```

Scilab code Exa 4.19 Calculate the capacitance per phase

```
//Calculate the capacitance per phase
clear;
clc;
//soltion
```

```
//given
Eo=8.85*10^-12//permitivity of the air
Vph=132*1000/sqrt(3);//phase voltage
d1=8;//m//distances
d2=6;//m
r=3*2.5*10^-3;//m//radius of conductor in m
C=4*%pi*Eo/log((2^(1/3))*(d1/r)*((d2^2+d1^2)/(4*d1^2+d2^2))^(1/3));
C_=C*100*1000;
printf("Capacitance of 100 km line= %.3f  f",C_*10^6);
```

Chapter 5

Performance of Short and Medium Transmission Lines

Scilab code Exa 5.1 Find voltage at sending end and percentage regulation and transmission efficiency

```
//Find voltage at sending end, percentage regulation
    and transmission efficiency
clear;
clc;
//soltion
//given
P=3300;//kW//power
Vr=33000;//kV//recieving voltage
pf=0.8;//peak factor
R=2;//ohm//resistance
X=3;//ohm//loop reactance
I=P*1000/(Vr*pf);
Vs=sqrt((Vr*pf+I*R)^2+((Vr*sind(acosd(pf)))+I*X)^2);
printf("Voltage at sending end(Vs)= %.3fV\n",Vs);
Pr=((Vs-Vr)*100)/Vr;
printf("Percentage regulation= %f percent\n",Pr);
Ll=I*I*R/1000;//line losses
nt=P*100/(P+Ll);
```

```
printf("Transmission efficiency= %.2f percent",nt)
```

Scilab code Exa 5.2 voltage at sending end and percentage regulation and total line losses and transmission efficiency

```
//voltage at sending end, percentage regulation,
   total line losses and transmission efficiency
clear;
clc;
//soltion
//given
P=5000;//kW//power
V=22000;//kV//recieving voltage
pf=0.8;//peak factor
R=4;//ohm//resistance
X=6;//ohm//loop reactance
Vr=V/sqrt(3);
I=P*1000/(3*round(Vr)*pf);
Vs=round(Vr)+(I*R*pf)+(I*X*sind(acosd(pf)));
Vsl=sqrt(3)*Vs;
printf("Sending end line voltage= %.3fkV\n",Vsl
   /1000)
Pr=((Vsl-V)*100)/V;
printf("Percentage regulation= %.2f percent\n",Pr);
Ll=3*(round(I))^2*R/1000;//line losses
printf("Total Line Losses= %.3fkW\n",Ll);
nt=P*100/(P+Ll);
printf("Transmission efficiency= %.3f percent",nt)
```

Scilab code Exa 5.3 find sending end voltage and regulation

```
//find sending end voltage and regulation
clear;
```

```
clc;
//soltion
//given
P=5000;//kW//power
V=11000;//kV//recieving voltage
pf=0.8;//peak factor
L=1.1*10^-3//H per km per phase//Line inductance
Ll=0.12*P*1000;
Vr=V/sqrt(3);
I=P*1000/(3*round(Vr)*pf);
R=Ll/(3*I^2);
X=5.1836;
Vs=round(Vr)+(round(I)*R*pf)+(I*X*sind(acosd(pf)));
Vsl=sqrt(3)*Vs;
printf("Line voltage at sending end= %.3f kV\n",Vsl
   /1000);
Pr=((Vsl-V)*100)/V;
printf("Percentage regulation= %.3f percent\n",Pr);
```

Scilab code Exa 5.4 Find sending end voltage and power factor and efficieny and regulation

```
//Find sending end voltage, power factor, efficieny
   and regulation
clear;
clc;
//soltion
//given
S=12000;//kVA//power supplied
pf=0.8;//power factor
del=1.73*10^-6;
d=140//cm//distance of the conductor
l=50*10^3;
Vrl=33000;//V//recieving end voltage
I=S*1000/(sqrt(3)*Vrl);
```

```
L1=0.15*S*1000*pf;
R=L1/(3*I*I);
a=del*l*100/(R);
r=sqrt(a/%pi);
re=r*exp(-1/4);
L=0.2*50*(10^-3)*log(d/re);
X=2*%pi*50*L;
X_=floor(X*100)/100;
Vs=Vrl/sqrt(3)+(I*R*pf)+(I*X_*sind(acosd(pf)));
Vsl=sqrt(3)*Vs;
printf("Sending end line voltage= %.4fkV\n",Vsl
   /1000)
spf=(Vrl*pf/sqrt(3)+I*R)/Vs;
printf("Sending end power factor= %.3f lagging\n",
   spf);
nt=S*pf*100/(S*pf+(L1/1000));
printf("Transmission efficiency= %.3f percent\n",nt)
Pr=((Vsl-Vrl)*100)/Vrl;
printf("Percentage regulation= %.3f percent\n",Pr);
```

Scilab code Exa 5.5 Find load end voltage and efficieny

```
//Find load end voltage and efficieny
clear;
clc;
//soltion
//given
P=3000//kW//output
Vsl=11000//volts
pf=0.8//lagging//power factor
R=3*0.4;//ohm//resistance of each conductor
X=3*0.8;//ohm//reactance of each conductor
Vs=Vsl/sqrt(3);
Z=(R*pf+X*sind(acosd(pf)));
Vs_=round(Vs);
```

```
printf("Vr=%d - %.1fI\n",Vs_,Z);
I_=P*1000/(3*pf)
Vr=poly(0,"Vr");
printf("I=%.0f/Vr\n",I_);
A=2.4*I_-Vs_*Vr+Vr^2
answ=roots(A);
Vr=5837.041;
Vrl=sqrt(3)*Vr;
printf("Line voltage at the end(Vrl)= %d V\n",Vrl);
I=I_/Vr;
Ll=3*I*I*R;
nt=P*1000*100/(P*1000+Ll);
printf("Transmission efficiency= %.1f percent",nt)
```

Scilab code Exa 5.6 Find current and voltage of sending end and percentage regulation and line losses and sending end power factor and transmission efficiency

```
//Find current and voltage of sending end, percentage regulation, line losses, sending end power factor and transmission efficiency
clear;
clc;
//soltion
//given
R=0.6125*100;//ohm//total resistance
X=1*100;//ohm//reactance
Y=17.5*10^-4;//S//total susecptance
Vr=66*1000;//V
pf=0.8;//power factor
P=20*10^6;//watts
Ir=(P/(Vr*pf))*complex(pf,-0.6);
Ic=complex(0,Y*Vr);
Is=Ir+Ic;
theta1=atand((imag(Is)/real(Is)));
```

```
printf("Sending end current= %.2f  %.3f A\n",abs(Is),theta1);
Vs=Vr+Is*(complex(R,X));
theta2=atand((imag(Vs)/real(Vs)));
printf("Sending end voltage= %.3f  %.2f Volts\n",abs(Vs),theta2);
phi=theta2-theta1;
printf("sending end power factor= %.3f(lag)\n",cosd(phi));
Pr=((abs(Vs)-Vr)*100)/Vr;
printf("Percentage regulation= %.1f percent\n",Pr);
Ll=(abs(Is))^2*R/1000;//line losses
printf("Total Line Losses= %.3fkW\n",Ll);
nt=P*100/(P+Ll*1000);
printf("Transmission efficiency= %.2f percent",nt)
```

Scilab code Exa 5.7 Find current and voltage of sending end and percentage regulation and transmission efficiency

```
//Find current and voltage of sending end and percentage regulation and transmission efficiency
clear;
clc;
//soltion
//given
R=0.2*150;//ohm//total resistance
X=0.5*150;//ohm//reactance
Y=150*3*10^-6;//S//total susecptance
Vrl=132*1000;//V
pf=0.8;//power factor
P=40*10^6;//MVA
Vr=Vrl/sqrt(3);
Ir_=(P/(sqrt(3)*Vrl));
Ir=Ir_*complex(pf,-0.6);
Z=complex(R,X);//ohm//Load impedance
```

```
V_=Vr+Ir*(Z/2);
Ic=V_*(%i)*Y;
Is=Ir+Ic;
theta1=atand((imag(Is)/real(Is)));
printf("Sending end current= %.3f  %.2f  A\n",abs(Is),theta1);
Vs=V_+Is*(Z/2);
theta2=atand((imag(Vs)/real(Vs)));
Vls=sqrt(3)*abs(Vs)/1000;
printf("Sending end line voltage= %.2fkV\n",Vls);
Pr=((abs(Vs)-Vr)*100)/Vr;
printf("Percentage voltage regulation= %.1f percent\n",Pr);
phi=theta2-theta1;
nt=(Vrl*Ir_*pf*100)/(Vls*1000*abs(Is)*cosd(phi));
printf("Transmission efficiency= %.2f percent",nt);
```

Scilab code Exa 5.8 Find current and voltage of sending end and percentage regulation

```
//Find current and voltage of sending end and percentage regulation
clear;
clc;
//soltion
//given
R=0.1425*200;//ohm//total resistance
X=0.49*200;//ohm//reactance
Y=8*10^-4;//S//total susecptance
Vrl=132*1000;//V
pf=0.8;//power factor
P=50*10^6;//MVA
Vr=round(Vrl/sqrt(3));
Ir_=(P/(sqrt(3)*Vrl));
Ir=Ir_*complex(pf,-0.6);
```

```
Icr=0.5*(%i*Y)*Vr;
Il=Ir+Icr;
Z=complex(R,X);//ohm//Load impedance
Vs=Vr+Il*(Z);
theta=atand((imag(Vs)/real(Vs)));
printf("Sending end voltage= %.3f  % .3f  \n",abs(Vs),theta);
Vls=sqrt(3)*abs(Vs)/1000;
printf("Sending end line voltage= %.2fkV\n",Vls);
M=1+0.5*(%i*Y)*Z;//THE BOOK HAS USED 0.9962 BUT IT IS 0.962
Vrlo=Vls/abs(M);
Pr=((Vrlo*1000-Vrl)*100)/Vrl;
printf("Percentage voltage regulation= %.1f percent\n",Pr);
//THE ANS OF THE REGULATION IS 21.4% BECAUSE OF TYPOLOGICAL ERROR
```

Scilab code Exa 5.9 Find current and voltage of sending end

```
//Find current and voltage of sending end
clear;
clc;
//soltion
//given
R=0.1*150;//ohm//total resistance
X=0.5*150;//ohm//reactance
Y=3*150*10^-6;//S//total susecptance
Vrl=110*1000;//V
pf=0.8;//power factor
P=50*10^6;//M watts
Vr=floor(Vrl/sqrt(3));
Ir_=(P/(sqrt(3)*Vrl*pf));
Ir=Ir_*complex(pf,-0.6);
Ic1=Vr*(%i*Y/2);
```

```
Il=Ir+Ic1;
Z=complex(R,X);
Vs=Vr+Il*Z;
theta=atand((imag(Vs)/real(Vs)));
Vls=sqrt(3)*abs(Vs)/1000;
printf("Sending end line voltage= %.2f kV\n",Vls);
Ic2=Vs*(%i*Y/2);
Is=Il+Ic2;
printf("Sending end current(Is)= %.1f A",abs(Is));
```

Scilab code Exa 5.10 Find regulation and charging current using nominal T method

```
//Find regulation and charging current using nominal
   T method
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the
   multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to
   polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

//given
P=50*10^6;//M watts
Vrl=132*1000;//V
pf=0.8//power factir
Vr=[floor(Vrl/sqrt(3)) 0];
```

```
Ir=[floor(P/(sqrt(3)*Vrl*pf)) -acosd(pf)];
A=[0.95 1.4];
B=[96 78];
C=[0.0015 90];
D=A;
Z1=rxr(A,Vr);
Z2=rxr(B,Ir);
AV=r2p(Z1);
BI=r2p(Z2);
Vs=AV+BI;
theta1=atand((imag(Vs)/real(Vs)));
printf("Sending end voltage= %.0f  %.2f   Volts\n", abs(Vs),theta1);
Y1=rxr(C,Vr);
Y2=rxr(D,Ir);
CV=r2p(Y1);
DI=r2p(Y2);
Is=CV+DI;
Ira=r2p(Ir);
Ic=Is-Ira;
theta2=atand(imag(Ic)/real(Ic));
Ic_=sqrt(round(imag(Ic))^2+round(real(Ic))^2);
printf("Charging current= %.1f  %f  A\n",Ic_, theta2);
Pr=((abs(Vs)/A(1)-Vr)*100)/Vr;
printf("Percentage regulation= %.0f percent\n",Pr);
//1. The Magnitude of Sending end voltage is 94066, it is due to rounding some of the values
//2. The angle in the book is 93.1  in charging current
```

Scilab code Exa 5.11 find sending end voltage and current and power and efficiency

```
//find sending end voltage and current and power and efficiency
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction
//given
P=50*10^6;//VA
Vrl=110*1000;//V
pf=0.8//power factir
Vr=[Vrl/sqrt(3) 0];
Ir=[P/(sqrt(3)*Vrl) -acosd(pf)];
A=[0.98 3];
B=[110 75];
C=[0.0005 80];
D=[0.98 3];
Z1=rxr(A,Vr);
Z2=rxr(B,Ir);
AV=r2p(Z1);
BI=r2p(Z2);
Vs=AV+BI;
theta1=atand((imag(Vs)/real(Vs)));
printf(”Sending end voltage= %.0f V\n”,abs(Vs));
Y1=rxr(C,Vr);
Y2=rxr(D,Ir);
CV=r2p(Y1);
DI=r2p(Y2);
Is=CV+DI;
```

```
theta2=atand(imag(Is)/real(Is));
printf("Magnitude of sending end current= %d A\n",abs(Is));
phis=theta2-theta1;
Ps=3*abs(Vs)*abs(Is)*cosd(phis);
printf("Sending end power=%.1fMW\n",floor(Ps/10^5)/10);
Pr=P*pf;
n=Pr*100/(floor(Ps/10^5)*10^5);
printf("Transmission Efficiency= %.1f percent",n);
//The value of voltage is 87427 V
```

Scilab code Exa 5.12 Find ABCD parameters and sending end voltage and current and power factor and transmission eficiency

```
//Find ABCD parameters and sending end voltage and current and power factor and transmission eficiency
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

//given
P=80*10^6;//watts
```

```
Vrl=220*1000;//V
pf=0.8//power factir
Vr=[Vrl/sqrt(3) 0];
Ir_=[P/(sqrt(3)*Vrl*pf) -acosd(pf)];
Ir=r2p(Ir_);
Z=[200 80];
Y=[0.0013 90];
a=rxr(Z,Y);
Ac=1+r2p(a)/2;
A=[abs(Ac) atand((imag(Ac)/real(Ac)))];
D=A;
printf("A=D= %.3f   %.1f    \n",A(1),A(2));
b=rxr(Z,Y);
Bc=1+r2p(b)/4;
B=[abs(Bc) atand((imag(Bc)/real(Bc)))];
B=rxr(Z,B);
printf("B= %.1f   %.2f    ohm\n",B(1),B(2));
C=Y;
printf("C=%.4f   %d    siemens\n",C(1),C(2));
Z1=rxr(A,Vr);
Z2=rxr(B,Ir_);
AV=r2p(Z1);
BI=r2p(Z2);
Vs=AV+BI;
theta1=atand((imag(Vs)/real(Vs)));
Vsl=sqrt(3)*abs(Vs);
printf("Sending end voltage= %dkV\n",round(Vsl/1000));
Y1=rxr(C,Vr);
Y2=rxr(D,Ir_);
CV=r2p(Y1);
DI=r2p(Y2);
Is=CV+DI;
theta2=atand(imag(Is)/real(Is));
printf("Sending end current= %.1f   %.1f  A \n",abs(Is),theta2);
phis=theta2-theta1;
Ps=3*abs(Vs)*abs(Is)*cosd(phis);
```

```
printf("Sending end power=%.2fMW\n",Ps/10^6);
n=P*100/Ps;
printf("Transmission Efficiency= %.1f percent",n);
```

Scilab code Exa 5.13 find sending end voltage and current and power and efficiency

```
//find sending end voltage and current and power and efficiency
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction
//given
P=50*10^6;//VA
Vrl=110*1000;//V
pf=0.8//power factir
Vr=[Vrl/sqrt(3) 0];
Ir=[P/(sqrt(3)*Vrl) -acosd(pf)];
A=[0.98 3];
B=[110 75];
C=[0.0005 80];
D=[0.98 3];
Z1=rxr(A,Vr);
Z2=rxr(B,Ir);
```

```
AV=r2p(Z1);
BI=r2p(Z2);
Vs=AV+BI;
theta1=atand((imag(Vs)/real(Vs)));
printf("Sending end voltage= %.0f V\n",abs(Vs));
Y1=rxr(C,Vr);
Y2=rxr(D,Ir);
CV=r2p(Y1);
DI=r2p(Y2);
Is=CV+DI;
theta2=atand(imag(Is)/real(Is));
printf("Magnitude of sending end current= %d A\n",abs(Is));
phis=theta2-theta1;
Ps=3*abs(Vs)*abs(Is)*cosd(phis);
printf("Sending end power=%.1fMW\n",floor(Ps/10^5)/10);
Pr=P*pf;
n=Pr*100/(floor(Ps/10^5)*10^5);
printf("Transmission Efficiency= %.1f percent",n);
//The value of voltage is 87427 V
//this is same as ex 12 because of printing mistake in book
```

Scilab code Exa 5.14 Determine ABCD constant and sending end power factor

```
//Determine ABCD constant and sending end power factor
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication in rectangular form
```

```
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [z]=rdr(A,B)//Function for the division in rectangular form
    z(1)=A(1)/B(1)
    z(2)=A(2)-B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

//given
P=100*10^6;//watts
Vrl=132*1000;//V
pf=0.8//power factir
Vr=[Vrl/sqrt(3) 0];
Ir=[P/(sqrt(3)*Vrl*pf) -acosd(pf)];
A=[0.98 1];
B=[100 75];
C=[0.0005 90];
D=A;
AB=rxr(A,B);
Ap=rdr(AB,B);
printf("A(in parallel)= D = %.2f  %d  \n",Ap(1),Ap(2));
BB=rxr(B,B);
Bp_=rdr(BB,B);
Bp=[Bp_(1)/2 Bp_(2)];//Bp is a half vector of the Bp_
printf("B(in parallel)= %d  %d  ohm\n",Bp(1),Bp(2));
printf("Here A1=A2 & D1=D2 therefore \n");
Cp=[C(1)*2 C(2)];
printf("C(in parallel)= %.3f  %d  siemens\n",Cp(1)
```

```
    ,Cp(2));
Z1=rxr(Ap,Vr);
Z2=rxr(Bp,Ir);
AV=r2p(Z1);
BI=r2p(Z2);
Vs=AV+BI;
theta1=atand((imag(Vs)/real(Vs)));
Y1=rxr(Cp,Vr);
Y2=rxr(Ap,Ir); //D = A
CV=r2p(Y1);
DI=r2p(Y2);
Is=CV+DI;
theta2=atand(imag(Is)/real(Is));
phis=theta1-theta2;
Spf=cosd(phis);//Sending end power factor
printf("Sending end power factor= %.3f(lagging)",Spf
    );
```

Chapter 6

Performance of Long Transmission Lines

Scilab code Exa 6.1 Determine auxiliary constants

```
//Determine auxiliary constants
clear;
clc;

//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication of rectangular
   z(1)=A(1)*B(1)
   z(2)=A(2)+B(2)
   endfunction

function [a]=r2p(z)//Function for rectangular to polar
   a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

function [v]=p2r(q)//Function for polar to rectangular
```

```
    v(1)=abs(q)
    v(2)=atand(imag(q)/real(q))
endfunction

//given
r=0.25;//ohm
x=0.48;//ohm
g=4*10^-9;//mho
b=2.53*10^-6;//mho
f=50;//frequency
l=1000;
z=complex(r,x);
y=complex(g,b);
Z_=z*1000;
Y_=y*1000;
Z=p2r(Z_);
Y=p2r(Y_);
YZ=rxr(Z,Y);
Y2Z2=rxr(YZ,YZ);
[Y3Z3]=rxr(Y2Z2,YZ);
A_=1+(r2p(YZ))/2+(r2p(Y2Z2))/24+(r2p(Y3Z3))/720;
A=p2r(A_);
printf("A = D = %.4f  %.2f  \n",A(1),A(2));
P_=(1+(r2p(YZ))/6+(r2p(Y2Z2))/120+(r2p(Y3Z3))/5040);
P=p2r(P_);
B=rxr(Z,P);
printf("B= %.2f  %.2f   ohm\n",B(1),B(2));
C=rxr(Y,P);
printf("C= %.2f*10^-3    %.2f    siemens\n",C(1)
   *1000,C(2));
```

Scilab code Exa 6.2 Determine sending end voltage and current

```
//Determine sending end voltage and current
clear;
```

```
clc;

//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

function [v]=p2r(q)//Function for polar to rectangular
    v(1)=abs(q)
    v(2)=atand(imag(q)/real(q))
endfunction


//given
P=80*10^6;//MW
Vrl=220*1000;//V
pf=0.8//power factir
Vr=[Vrl/sqrt(3) 0];
Ir=[P/(sqrt(3)*Vrl*pf) -acosd(pf)];
Z=[200 80];
Y=[0.0013 90];
YZ=rxr(Z,Y);
Y2Z2=rxr(YZ,YZ);
[Y3Z3]=rxr(Y2Z2,YZ);
A_=1+(r2p(YZ))/2+(r2p(Y2Z2))/24+(r2p(Y3Z3))/720;
A=p2r(A_);
printf("A = D = %.4f  % .2f  \n",A(1),A(2));
P_=(1+(r2p(YZ))/6+(r2p(Y2Z2))/120+(r2p(Y3Z3))/5040);
P=p2r(P_);
```

```
B=rxr(Z,P);
printf("B= %.2f  %.2f  ohm\n",B(1),B(2));
C=rxr(Y,P);
printf("C= %.6f  %.2f  siemens\n",C(1),C(2));
D=A;
Z1=rxr(A,Vr);
Z2=rxr(B,Ir);
AV=r2p(Z1);
BI=r2p(Z2);
Vs=AV+BI;
theta1=atand((imag(Vs)/real(Vs)));
printf("Sending end voltage= %.3fkV\n",sqrt(3)*abs(Vs)/1000);
Y1=rxr(C,Vr);
Y2=rxr(D,Ir);
CV=r2p(Y1);
DI=r2p(Y2);
Is_=CV+DI;
Is=p2r(Is_);
printf("Magnitude of sending end current= %.1f  %.2f  A\n",Is(1),Is(2));
```

Scilab code Exa 6.3 Determine percentage rise in voltage

```
//Determine percentage rise in voltage
clear;
clc;
//soltion
//given
f=50;//Hz//frequency
w=2*%pi*f;
l=200;//km//length
RiV=((w^2)*(l^2)*10^-8)/18;
printf("Rise in voltage= %.2f percent",RiV);
```

Scilab code Exa 6.4 alculate constants of equivalent circuit of line

```
//calculate constants of equivalent circuit of line
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the
    multiplication of rectangular
     z(1)=A(1)*B(1)
     z(2)=A(2)+B(2)
     endfunction

function [a]=r2p(z)//Function for rectangular to
    polar
     a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

//given
P=100*10^6;//VA
Vrl=220*1000;//V
Zse_=complex(1,6);
Zseo=(Zse_*(Vrl^2))/(P*100);
Zse=[abs(Zseo) atand(imag(Zseo)/real(Zseo))];
A=[1 0.8];
B=[169.52 84.6];
C=[0.00135 90];
D=A;
CZ=rxr(C,Zse);
Ao_=r2p(A)+r2p(CZ);
Ao=[abs(Ao_) atand(imag(Ao_)/real(Ao_))];
printf("Ao = %.5f  %.2f  \n",Ao(1),Ao(2));
DZ=rxr(D,Zse);
Bo_=r2p(B)+r2p(DZ);
```

```
Bo=[abs(Bo_) atand(imag(Bo_)/real(Bo_))];
printf("Bo = %.2f  % .2f   ohm\n",Bo(1),Bo(2));
Co=C;
Do=A;
printf("Co = %.5f  %d   siemens\n",Co(1),Co(2));
printf("Do =  %d  % .1f  ",Do(1),Do(2));
//the value of Ao is different because book has a calculation mistake and according to book it is 0.9799  11.49
```

Scilab code Exa 6.5 calculate constants of equivalent circuit of line

```
//calculate constants of equivalent circuit of line
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication in rectangular form
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
endfunction

function [z]=rdr(A,B)//Function for the division in rectangular form
    z(1)=A(1)/B(1)
    z(2)=A(2)-B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

//given
```

```
Zse=[12 80];
A=[0.9 1];
B=[26 68];
D=A;
AD_=r2p(rxr(A,D))-1;
AD=[abs(AD_) atand(imag(AD_)/real(AD_))];
C=rdr(AD,B);
CZ=rxr(C,Zse);
Ao_=r2p(A)+r2p(CZ);
Ao=[abs(Ao_) atand(imag(Ao_)/real(Ao_))];
printf("Ao = %.4f  %.2f  \n",Ao(1),Ao(2));
DZ=rxr(D,Zse);
CZ2=rxr(CZ,Zse)
Bo_=r2p(B)+2*r2p(DZ)+r2p(CZ2);
Bo=[abs(Bo_) atand(imag(Bo_)/real(Bo_))];
printf("Bo = %.2f  %.2f   ohm\n",Bo(1),Bo(2));
Co=C;
Do=Ao;
printf("Co = %.5f  %d   siemens\n",Co(1),Co(2));
printf("Do = %.4f  %.2f  ",Do(1),Do(2));
//there is a mistake in the value of C(=0.00738  2.55  ) so all the values are changed
```

Scilab code Exa 6.6 calculate Ao and Bo and Co and Do constants

```
//calculate Ao and Bo and Co and Do constants
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication in rectangular form
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
endfunction
```

```
function [z]=rdr(A,B)//Function for the division in rectangular form
    z(1)=A(1)/B(1)
    z(2)=A(2)-B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

function[a]=p2r(z)//Funtion for polar to rectangular
    a(1)=abs(z);
    a(2)=atand(imag(z)/real(z));
endfunction

//given
Zt=[100 70];
Yt=[0.0002 -75];
A=[0.92 5.3];
B=[65.3 81];
D=A;
AD_=r2p(rxr(A,D))-1;//A*D-1
AD=[abs(AD_) 180+atand(imag(AD_)/real(AD_))];
C=rdr(AD,B);//(A*D-1)/B
BYt=rxr(Yt,B);
CZt=rxr(C,Zt);
YtZt_=r2p(rxr(Yt,Zt))*2+1;//1+2*Yt*Zt
P=[abs(YtZt_) atand(imag(YtZt_)/real(YtZt_))];//Let P=1+2*Yt*Zt
YtZto=r2p(rxr(Yt,Zt))+1;//1+Yt*Zt
Q=[abs(YtZto) atand(imag(YtZto)/real(YtZto))];//Let Q=1+Yt*Zt
Ao_=r2p(rxr(A,P))+r2p(BYt)+r2p(rxr(CZt,Q));//A*(1+2*Yt*Zt)+B*Yt+C*Zt(1+Yt*Zt)
Ao=[abs(Ao_) atand(imag(Ao_)/real(Ao_))];
printf("Ao = %.4f  %.2f \n",Ao(1),Ao(2));
```

```
DZt=rxr(D,Zt);//D*Zt
CZt2=rxr(CZt,Zt);//C*Zt^2
Bo_=r2p(B)+2*r2p(DZt)+r2p(CZt2);//2*A*Zt+B+C*Zt^2
Bo=[abs(Bo_) atand(imag(Bo_)/real(Bo_))];
printf("Bo = %.2f  %.2f   ohm\n",Bo(1),Bo(2));
BYt2=r2p(rxr(BYt,Yt));//B(Yt^2)
AYt=rxr(A,Yt);//A*Yt
AYt_YZt=rxr(p2r(2*r2p(AYt)),p2r(1+YtZto)/2);//2*A*Yt(1+Y*Zt)
YtZt2=rxr(Q,Q);//(1+Yt*Zt)^2
Co_=r2p(AYt_YZt)+BYt2+r2p(rxr(C,YtZt2));//2*A*Yt(1+Y*Zt)+B*Yt^2+C*(1+Yt*Zt)^2
Co=[abs(Co_) atand(imag(Co_)/real(Co_))];;
Do=Ao;
printf("Co = %.4f  %.1f    siemens\n",Co(1),Co(2));
printf("Do = %.4f  %.2f  ",Do(1),Do(2));
```

Scilab code Exa 6.7 calculate equivalent T and pi constants

```
//calculate equivalent T &    constants
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication in rectangular form
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
endfunction

function [z]=rdr(A,B)//Function for the division in rectangular form
    z(1)=A(1)/B(1)
    z(2)=A(2)-B(2)
    endfunction
```

```
function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

function[a]=p2r(z)//Funtion for polar to rectangular
    a(1)=abs(z);
    a(2)=180+atand(imag(z)/real(z));
endfunction

//given
A=[0.9 1];
B=[85 75];
C=[0.0013 91];
D=A;
Z=rdr(p2r(2*(r2p(A)-1)),C);
printf("Equivalent T network\n");
printf("Series Impedance Z=%.2f %.2f ohm\n",Z(1),Z(2));//IN BOOK Z=156.92 80.5 BECAUSE OF ROUNDING OFF THINGS
Y=C;
printf("Shunt Admitttance Y=%.4f %.0f siemens\n",Y(1),Y(2));
printf("Equivalent  network\n");
Zp=B;
Yp=rdr(p2r(2*(r2p(A)-1)),B);
printf("Series Impedance Z=%.f %.f ohm\n",Zp(1),Zp(2));
printf("Shunt Admitttance Y=%.4f %d siemens\n",Yp(1),Yp(2));
```

Scilab code Exa 6.8 find sending end reactive and active power

```
//find sending end reactive and active power
```

```
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction
//given
Sr=7.5*10^6;//VA
Vrl=32*1000;//V
pf=0.85//power factir
Vr=[Vrl/sqrt(3) 0];
Ir=[Sr/(sqrt(3)*Vrl) -acosd(pf)];
A=[1 0];
B=[11.18 63.43];
D=A;
C_=r2p(rxr(A,D))-1;
C=[abs(C_) 0]
AV=r2p(rxr(A,Vr));
BI=r2p(rxr(B,Ir));
Vs=AV+BI;
theta1=atand((imag(Vs)/real(Vs)));
printf("Sending end voltage= %.f  % .1f  V \n",abs(Vs),theta1);
Y1=rxr(C,Vr);
Y2=rxr(D,Ir);
CV=r2p(Y1);
DI=r2p(Y2);
Is=CV+DI;
theta2=atand(imag(Is)/real(Is));
printf("Sending end current= %.2f  % .1f   A\n",abs(
```

```
    Is),theta2);
phis=theta1-theta2;
Pa=3*abs(Vs)*abs(Is)*cosd(phis);//Active power
printf("Sending end power=%.2f MW\n",Pa/10^6);
Pr=3*abs(Vs)*abs(Is)*sind(phis);//Reactive power
printf("Reactive power= %.1f MVAR",Pr/10^6)
```

Scilab code Exa 6.9 find sending end voltage and regulation and recieving end rective and synchornous power

```
//find sending end voltage, regulation, recieving
    end rective and synchornous power
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the
    multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to
    polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction
//given
P=50*10^6;//VA
Vrl=110*1000;//V
pf=0.8;//power factir
Vr=[Vrl/sqrt(3) 0];
Ir=[P/(sqrt(3)*Vrl*pf) -acosd(pf)];
A=[0.96 1];
B=[100 80];
AV=r2p(rxr(A,Vr));
```

```
BI=r2p(rxr(B,Ir));
Vs=AV+BI;
theta1=atand((imag(Vs)/real(Vs)));
Vsl=sqrt(3)*abs(Vs);
printf("Sending end voltage= %.3fkV\n",Vsl/1000);
vr=(Vsl-Vrl)*100/Vrl;
printf("Voltage regulation= %.3f percent\n",vr);//IN
    BOOK IT IS 20.786%
clear;
Pr=70;//MW
Vsl=120;//kV
Vrl=110;//kV
A=0.96;
B=100;
bta=80;
alp=1;
b_d=acosd((70+(A/B)*Vrl^2*cosd(bta-alp))/(Vrl*Vsl/B)
    );//beta-del
Qr=Vrl*Vsl*sind(b_d)/B-(A/B)*Vrl^2*sind(bta-alp);
printf("Recieving end reactive power= %.2f MVAR\n",
    Qr);
Pc=Pr*tand(acosd(0.8))-Qr;
printf("Power delivered by synchronous generator= %
    .3f MVAR",Pc);
```

Scilab code Exa 6.10 find sending end voltage and charging current and power

```
//find sending end voltage, charging current, power
clear;
clc;
//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the
    multiplication of rectangular
```

```
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

function [z]=rdr(A,B)//Function for the division in rectangular form
    z(1)=A(1)/B(1)
    z(2)=A(2)-B(2)
    endfunction

//given
Vrl=230*1000;//V
Vs=[Vrl/sqrt(3) 0];
Ir=[0 0];
A=[0.938 1.2];
B=[131.2 72.3];
C=[0.001 90];
Vr_=r2p(rdr(Vs,A));
theta1=atand((imag(Vr_)/real(Vr_)));
Vr=[abs(Vr_) theta1];
Vrl=sqrt(3)*abs(Vr_);
printf("Sending end voltage= %.1fkV\n",Vrl/1000);
Ic=rxr(C,Vr);
printf("Line charging current= %.2f  % .1f A \n",Ic(1),Ic(2));
Vrl_=220;//kV
Vsl=230;//kV
Pr=Vrl_*Vsl/B(1)-(A(1)/B(1))*(Vrl_^2)*(cosd(B(2)-A(2)));//IN BOOK IT IS 272.58 MW DUE TO TYPOLOGICAL ERROR
printf("Maximum power transmitted= %f MW\n",Pr);
Qr=(A(1)/B(1))*(Vrl_^2)*(sind(B(2)-A(2)));
printf("Recieving reactive power required at
```

```
recieving end= %.2f MVAR",Qr);
```

Scilab code Exa 6.11 Determine sending end voltage and current and power factor and MVA and power angle

```
//Determine sending end voltage and current and power factor and MVA and power angle
clear;
clc;

//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

function [v]=p2r(q)//Function for polar to rectangular
    v(1)=abs(q)
    v(2)=atand(imag(q)/real(q))
endfunction

//given
P=40*10^6;//MVA
Vrl=220*1000;//V
pf=0.8//power factir
Vr=[Vrl/sqrt(3) 0];
Ir=[P/(sqrt(3)*Vrl) -acosd(pf)];
```

```
z=complex(0.105,0.3768)*500;
Z=[floor(abs(z)*1000)/1000 atand(imag(z)/real(z))];
y=complex(0,2.882*10^-6)*500;
Y=[abs(y) 90];
YZ=rxr(Z,Y);
Y2Z2=rxr(YZ,YZ);
A_=1+(r2p(YZ))/2+(r2p(Y2Z2))/24;
A=p2r(A_);
P_=(1+(r2p(YZ))/6+(r2p(Y2Z2))/120);
P=p2r(P_);
B=rxr(Z,P);
C=rxr(Y,P);
D=A;
AV=r2p(rxr(A,Vr));
BI=r2p(rxr(B,Ir));
Vs=AV+BI;
theta1=atand((imag(Vs)/real(Vs)));
Vsl=sqrt(3)*abs(Vs)/1000;
printf("Sending end voltage= %.3fkV\n",Vsl);
CV=r2p(rxr(C,Vr));
DI=r2p(rxr(D,Ir));
Is_=CV+DI;
Is=p2r(Is_);
theta2=Is(2);
printf("Magnitude of sending end current= %.1f A\n",Is(1));
Spf=cosd(theta2-theta1);
printf("Sending end power factor= %.3f leading\n",Spf);
Ps=sqrt(3)*Vsl*Is(1)/1000;
printf("Sending end MVA= %.2f MVA\n",Ps);
printf("Power angle= %.3f  ",theta1);
//ALL THE ANS ARE DIFFRENT BECAUSE OF ROUND OFF IN THE BOOK
```

Scilab code Exa 6.12 Find sending end voltage and current and power factor

```
//Find sending end voltage and current and power
   factor
clear;
clc;

//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the
   multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
endfunction

function [a]=r2p(z)//Function for rectangular to
   polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

function [v]=p2r(q)//Function for polar to
   rectangular
    v(1)=abs(q)
    v(2)=atand(imag(q)/real(q))
endfunction

//given
P=40*10^6;//MVA
Vrl=220*1000;//V
Vr=[Vrl/sqrt(3) 0];
Ir=[0 0];
z=complex(0.105,0.3768)*500;
Z=[floor(abs(z)*1000)/1000 atand(imag(z)/real(z))];
y=complex(0,2.882*10^-6)*500;
Y=[abs(y) 90];
YZ=rxr(Z,Y);
Y2Z2=rxr(YZ,YZ);
```

```
A_=1+(r2p(YZ))/2+(r2p(Y2Z2))/24;
A=p2r(A_);
P_=(1+(r2p(YZ))/6+(r2p(Y2Z2))/120);
P=p2r(P_);
C=rxr(Y,P);
D=A;
AV=rxr(A,Vr);
Vs=AV;
Vsl=sqrt(3)*Vs(1)/1000;
printf("Sending end voltage= %.3f kV\n",Vsl);//IN BOOK DUE TO PRINTING MISTAKE IT IS 119.51 kV
Is=rxr(C,Vr);
printf("Sending end line current= %.1f A\n",Is(1));//IN BOOK IT IS 171.4 A DUE TO ROUND OFF
Spf=cosd(Vs(2)-Is(2));
printf("Sending end power factor= %.4f leading",Spf);
```

Scilab code Exa 6.13 Find characteristics impedance and propogation constant and ABCD constants

```
//Find characteristics impedance and propogation constant and ABCD constants
clear;
clc;
//soltion
//FUNCTIONS

function [z]=rxr(A,B)//Function for the multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
endfunction

function [z]=rdr(A,B)//Function for the division in
```

```
rectangular form
    z(1)=A(1)/B(1)
    z(2)=A(2)-B(2)
    endfunction

function [v]=p2r(q)//Function for polar to rectangular
    v(1)=abs(q)
    v(2)=atand(imag(q)/real(q))
endfunction

//given
Z=complex(14.1,51.48);
Y=complex(0,1.194*10^-3);
l=200;//length of the line
z=Z/l;
y=Y/l;
Zc=p2r(sqrt(z/y));
printf("Characteristics Impedance= %d %.2f  ohm\n",ceil(Zc(1)),Zc(2));
P=sqrt(z*y);//propogation constant
printf("Propagation constants= %f + i%f\n",real(P),imag(P));
al=real(P)*l;
bl=imag(P)*l;
yl=P*l;
A=p2r(cosh(yl));
printf("A = D = %.4f  %.2f  \n",A(1),A(2));
B=rxr(Zc,p2r(sinh(yl)));
printf("B= %.2f  %.2f  ohm\n",B(1),B(2));
C=rdr(p2r(sinh(yl)),Zc);
printf("C= %.6f  %.2f  mho\n",C(1),C(2));
```

Scilab code Exa 6.14 Determine recieving end voltage and current

```
//Determine recieving end voltage and current
clear;
clc;

//soltion
//FUNCTIONS
function [z]=rxr(A,B)//Function for the
   multiplication of rectangular
    z(1)=A(1)*B(1)
    z(2)=A(2)+B(2)
    endfunction

function [a]=r2p(z)//Function for rectangular to
   polar
    a=z(1)*complex(cosd(z(2)),sind(z(2)))
endfunction

function [v]=p2r(q)//Function for polar to
   rectangular
    v(1)=abs(q)
    v(2)=atand(imag(q)/real(q))
endfunction

//given
P=60*10^6;//MW
Vsl=220*1000;//V
Vs=Vsl/sqrt(3);
pf=0.8//power factir
Z=[200 80];
Y=[0.0013 90];
YZ=rxr(Z,Y);
Y2Z2=rxr(YZ,YZ);
A_=1+(r2p(YZ))/2+(r2p(Y2Z2))/24;
A=p2r(A_);
printf("A = D = %.3f  %.3f  \n",A(1),A(2));
P_=(1+(r2p(YZ))/6+(r2p(Y2Z2))/120);
B=rxr(Z,p2r(P_));//IN BOOK IT'S 1941.56 DUE TO
   TYPOLOGICAL ERROR
```

```
printf("B= %.2f  %.2f   ohm\n",B(1),B(2));
D=A;
Vr_=poly(0, 'Vr');
Ir=[P/(3*pf) -acosd(pf)];
C1=A(1);//constant of A*Vr
C2=B(1)*Ir(1);//constant of B*I
BI_ang=B(2)+Ir(2);//angle between B and I
BI= C2*(cosd(BI_ang)+%i*sind(BI_ang));
AV= C1*(cosd(1.41)+%i*sind(1.41));//1.41= Angle between A and V
com=numer(((real(AV)*Vr_+real(BI)/Vr_)^2+(imag(AV)*Vr_+imag(BI)/Vr_)^2-Vs^2));//considering only numerator part
Vr=roots(com);
Vr1=99746;//by selecting the positive value & near to sending end voltage
Vrl=sqrt(3)*Vr1/1000;
printf("Receiving end line voltage= %.2f kV\n",Vrl);
Irl=Ir(1)/Vr1;
printf("Receiving end line current= %.0f A",Irl);
```

Scilab code Exa 6.15 Determine the induced voltage in the telephone line

```
//Determine the induced voltage in the telephone line
clear;
clc;

//soltion
//given
Vl=132*1000;//Volt
P=28*10^6;//load in kw
pf=0.85;//power factor
f=50;//Hz
l=200;//length of the line
```

```
r=0.005;//radius of conductor
hA=20;//height of the line
Ao=4*sqrt(3)/2;
dAP=Ao+5;
dAQ=dAP+1;;
dBP=sqrt(5*5+2*2);
dBQ=sqrt(6*6+2*2);
Ma=0.2*log(dAQ/dAP);
Mb=0.2*log(dBQ/dBP);
M=(Mb-Ma)*10^-3;
I=P/(sqrt(3)*Vl*pf);
Vm=2*%pi*f*M*I;
printf("For 200 km line induced voltage= %.1f volts\n",Vm*l);
Va=Vl/sqrt(3);
Vb=Va;
Vpa=Va*log((2*(hA+Ao)-dAP)/dAP)/log((2*(hA+Ao)-r)/r);
Vpb=Vb*log((2*(hA+Ao)-dBP)/dBP)/log((2*(hA+Ao)-r)/r);
Vp=Vpb-Vpa;
printf("The potential of telephone line= %d volts",Vp);
//the ans in the book is Vm= 90.4 volts and Vp=4396 because of using round off in some values
```

Chapter 7

Corona

Scilab code Exa 7.1 Determine line voltage for commencing of corona

```
//Determine line voltage for commencing of corona
clear;
clc;
//soltion
//given
d=3;//m
r=2;//cm
go=30/sqrt(2);//kV/cm.... Dielectric strength of air
Vdo=go*r*log(d*100/r);
Vl=sqrt(3)*Vdo;
printf("Line voltage for corona formation= %.2f kV",
    Vl);
//In book its 209.53 kV because of some typological
    error
```

Scilab code Exa 7.2 Determine whether corona will be there or not

```
//Determine whether corona will be there or not
```

```
clear;
clc;
//soltion
//given
Er=4;//relative permittivity
r=3.52/2;//cm
Vp=28;//kV//Voltage between conductor and an earthed
    clamp surrounding the porcelain
g1=poly(0,"g1");
r1=4/2;//cm
r2=10/2;//cm
g2=r*g1/(Er*r1);
g1max=roots(g1*r*log(r1/r)+g2*r1*log(r2/r1)-28);
printf("Maximum gradient on conductor surface= %.3f
   kV/cm\n",g1max);
printf("If gradient exceeds dielectric strength of
   air (21.1kV/cm) the corona will be present \n In
   this case it is present");
```

Scilab code Exa 7.3 Determine critical discruptive voltage for line

```
//Determine critical discruptive voltage for line
clear;
clc;
//soltion
//given
d=2*100;//cm
r=0.5;//cm
go=30/sqrt(2);//kV/cm.... Dielectric strength of air
mo=0.8;//Irregularity factor
del=0.95//air density factor
Vdo=mo*go*del*r*log(d/r);
Vl=sqrt(3)*Vdo;
printf("Line voltage (R.M.S)= %.2f kV",Vl);
```

Scilab code Exa 7.4 Find spacing between the conductor

```
//Find spacing between the conductor
clear;
clc;
//soltion
//given
r=1;//cm
go=30/sqrt(2);//kV/cm.... Dielectric strength of air
mo=1;//Irregularity factor
del=1//air density factor
Vdo=220/sqrt(3);
d=exp(Vdo/(mo*go*del*r));
printf("Spacing between the conductor (d)= %.2f m",d
   /100)
```

Scilab code Exa 7.5 Determine critical discruptive voltage for line

```
//Determine critical discruptive voltage for line
clear;
clc;
//soltion
//given
d=2*100;//cm
r=1.2;//cm
go=30/sqrt(2);//kV/cm.... Dielectric strength of air
mo=0.96;//Irregularity factor
b=72.2;//barometric pressure
t=20;//temperature
del=3.92*b/(273+t);//air density factor
Vdo=mo*go*del*r*log(d/r);
Vl=sqrt(3)*Vdo;
```

```
printf("Line voltage (R.M.S)= %.2f kV",Vl);
//In book its 208 kV because of rounding of floating points
```

Scilab code Exa 7.6 Determine critical discruptive voltage for line and corona loss

```
//Determine critical discruptive voltage for line and corona loss
clear;
clc;
//soltion
//given
Vph1=106/sqrt(3);//kV
Pc1=54;//kW//loss at Vph1
Vph2=110/sqrt(3);//kV
Pc2=95;//kW//loss at Vph2
Vphu=115/sqrt(3);//kV
printf("Pc   (Vph-Vdo)^2\n");
Vdo=poly(0,"Vdo");
A=roots((Vph1-Vdo)^2*Pc2-(Vph2-Vdo)^2*Pc1);
Vdo=54.123123;//after the solution of roots
Pcu=Pc1*((Vphu-Vdo)^2)/((Vph1-Vdo)^2)
printf("Corona loss at 115 kV= %.2f kW\n",Pcu);
printf("Critical discruptive voltage= %.2f kV",sqrt(3)*Vdo);
```

Scilab code Exa 7.7 Determine critical disruptive voltage and Visual critical voltage and Corona loss

```
//Determine critical disruptive voltage and Visual critical voltage and Corona loss
clear;
```

```
clc;
//soltion
//given
r=1.036/2;//cm//conductor radius
d=2.44*10^2;//cm//spacing
go=21.1;//kV/cm//Dielectric strength
mo=0.85;//irregularity factor
mv=0.72;//roughness factor
b=73.15;//pressure
t=26.6;//temperature
f=50;//frequency
del=3.92*b/(273+t);
Vph=110/sqrt(3);//kV//Voltage to which conductor are subjected
Vdo=go*del*mo*r*log(d/r);
Vvo=go*del*mv*r*(1+0.3/sqrt(del*r))*log(d/r);
printf("Critical voltage to neutral= %.3f kV(rms)\n",Vdo);
printf("Visual critical voltage to neutral= %.1f kV(rms)\n",Vvo);
Pc=(244/del)*(f+25)*sqrt(r/d)*(Vph-0.8*Vdo)^2*10^-5;
printf("Total corona loss for 160 km, 3 phase line= %d kW\n",ceil(160*3*Pc));
ra=Vph/(0.8*Vdo);
k=0.46;
printf("For this value of the Vph/Vdo(%.2f) K= %.2f\n",ra,k);
Pc2=21*10^-6*f*((Vph)^2)*k/(log10(d/r))^2;
printf("Total corona loss(under bad weather) for 160 km, 3 phase line= %.2f kW\n",160*3*Pc2);
//IN BOOK THE CORONA LOSS UNDER BAD CONDITION IS 1308.5 BECAUSE OF SOME TYPOLOGICAL ERROR
```

Scilab code Exa 7.8 Find corona characteristics

```
//Find corona characteristics
clear;
clc;
//soltion
//given
r=1/2;//cm//conductor radius
d=3*10^2;//cm//spacing
go=21.1;//kV/cm//Dielectric strength
mo=0.85;//irregularity factor
mv=0.72;//roughness factor
mv_=0.82;//for general corona
b=74;//pressure
t=26;//temperature
f=50;//frequency
del=3.92*b/(273+t);
Vph=110/sqrt(3);//kV//Voltage to which conductor are subjected
Vdo=go*del*mo*r*log(d/r);
Vvo_=go*del*mv*r*(1+0.3/sqrt(del*r))*log(d/r);
Vvo=Vvo_*(mv_/mv);
printf("Critical voltage to neutral= %.2f kV(rms)\n",Vdo);
printf("Visual critical voltage to neutral= %.1f kV(rms)\n",Vvo);
Pc=(244/del)*(f+25)*sqrt(r/d)*(Vph-Vdo)^2*10^-5;
printf("Total corona loss for 175 km, 3 phase line= %d kW\n",ceil(175*3*Pc));
Pc_=(244/del)*(f+25)*sqrt(r/d)*(Vph-0.8*Vdo)^2*10^-5;
printf("Total corona loss for 175 km, 3 phase line= %d kW\n",ceil(175*3*Pc_));
//THE ANS IN BOOK OF FAIR AND STORMY CONDITION IS 253 kW AND 1464kW BECAUSE OF USING ROUND OFF VALUES
```

Chapter 8

Insulators

Scilab code Exa 8.1 find voltage distribution across each insulator and string efficiency

```
//find voltage distribution across each insulator
   and string efficiency
clear;
clc;
//soltion
//given
k=1/6;//ratio
V=poly(0,"V");
V1=100/(k^3+6*k^2+10*k+4);
V2=(1+k)*V1;
V3=(1+3*k+k*k)*V1;
V4=(1+6*k+5*k^2+k^3)*V1;
printf("V1= %.2f percent of V\n V2= %.2f percent of
   V\n V3= %.2f percent of V\n V4= %.2f percent of V
   \n",V1,V2,V3,V4);
se=100*100/(4*V4);
printf("String efficiency= %.1f",se);
```

Scilab code Exa 8.2 find max safe working voltage and string efficiency

```
//find max safe working voltage and string
    efficiency
clear;
clc;
//soltion
//given
k=0.08;//ratio
V3=15;
V1=V3/(1+3*k+k*k);
V2=V1*(1+k);
V=V1+V2+V3;
printf("Max and safe working voltage= %.2f kV\n",V);
Se=V*100/(3*V3);
printf("String efficiency %.2f percent",Se);
```

Scilab code Exa 8.3 find ratio of ground to mutual capacitance and system line voltage and string efficiency

```
//find ratio of ground to mutual capacitance, system
    line voltage and string efficiency
clear;
clc;
//soltion
//given
V2=20;//kV
V3=25;//kV
k=poly(0,"k");
k=roots(V2*(1+3*k+k*k)-V3*(1+k));
k=0.13;//Considering only positive part
V1=V2/(1+k);
V=V1+V2+V3;
Se=V*100/(3*V3);
printf("K =%.2f \nSystem line voltage(V)= %.3fkV \
```

```
   nString Efficiency= %.1f percent",k,V,Se);
```

Scilab code Exa 8.4 find system line voltage and string efficiency

```
//find system line voltage and string efficiency
clear;
clc;
//soltion
//given
V2=20;//kV
V3=30;//kV
k=poly(0,"k");
k=roots(V2*(1+3*k+k*k)-V3*(1+k));
k=0.28;//Considering only positive part
V1=V2/(1+k);
V4=V1*(1+6*k+5*k^2+k^3);
V=V1+V2+V3+V4;
Se=V*100/(4*V4);
printf("System line voltage(V)= %.3f kV \nString
   Efficiency= %.3f percent",sqrt(3)*V,Se);
```

Scilab code Exa 8.5 find max safe working voltage

```
//find max safe working voltage
clear;
clc;
//soltion
//given
V3=11;//kV
k=12.5/100;//shunt/self cpacitance
V=poly(0,"V");
V1=V/(3+4*k+k*k);
V3_=V1*(1+3*k+k*k);
```

```
V=roots(V3-V3_);
printf("Maximum Voltage for string= %.2fkV",V);
```

Scilab code Exa 8.7 Find the values of line to pin capacitance

```
//Find the values of line to pin capacitance
clear;
clc;
//soltion
//given
n=5;
for p = 1:4
    C=p/(n-p);
    printf("C%d = %.3f*C\n",p,C);
end
```

Scilab code Exa 8.8 find string efficiency

```
//find string efficiency
clear;
clc;
//soltion
//given
V1=poly(0,"V1");
V2=poly(0,"V2");
V3=poly(0,"V3");
V=poly(0,"V");
//Since wC is common so its cancelled
function x=%c_sign(a)
            x = '1.05*V2/1.2' + a + '0.05*V3/1.2'
            endfunction
printf("V1=");
disp(sign('+'));
```

```
function x=%c_sin(a)
            x = '-1.2*V2/0.2' + a + '1.05*V3/0.2'
            endfunction
printf("\n\nV1=");
disp(sin('+'));
V2=25/23.25*V1;
V3=1.65/1.1625*V1;
printf("\n\nOn solving above equation\n\nV2= ");
disp(V2);
printf("V3=");
disp(V3);
V=V1+V2+V3;
V1=roots(1-V);//ignoring 'V' for making calculation easy
V2=25/23.25*V1*poly(0,"V");
V3=1.65/1.1625*V1;
Se=100/(3*V3);
printf("\n\nString Efficiency= %.1f percent",Se);
```

Scilab code Exa 8.9 Calculate voltage on line end unit and capacitance Cx required

```
//Calculate voltage on line end unit and capacitance Cx required
clear;
clc;
//soltion
//given
V=20;//kV
Sc=0.2;//shunt capacitance
V2=poly(0,"V2");
V1=poly(0,"V1");
V2=V1+0.2*V1;
V3=V2;
v1=roots(V1+V2+V3-V);
```

```
V3=1.2*v1;
printf("Voltage on line end unit= V3= %.2f kV\n",V3);
Cx=poly(0,"Cx");
C=poly(0,"C");
printf("For Cx \nC*1.2*V1 + Cx*1.2*V1 = C*1.2*V1 + C*2.2V1/5\n");
C_=roots(Cx*1.2-2.2/5);//after simplifying the eqution
Cx=C_*C;
printf("Cx=");
disp(Cx);
printf("Farads")
```

Chapter 9

Underground Cables

Scilab code Exa 9.1 Determine insulation resistance

```
//Determine insulation resistance
clear;
clc;
//soltion
//given
row=4*10^12;//ohm m
l=3*10^3;//m
r1=12.5;//mm
r2=12.5+5;//mm
Rins=row*log(r2/r1)/(2*%pi*l);
printf("Insulation resistance (Rins)= %.1f M ohm",
   Rins/10^6);
```

Scilab code Exa 9.2 Determine resistivity of dielectric in a cable

```
//Determine resistivity of dielectric in a cable
clear;
clc;
```

```
//soltion
//given
Rins=1840*10^6;//ohm
l=2*10^3;//m
r1=2/2;//mm
r2=6/2;//mm
row=Rins*(2*%pi*l)/log(r2/r1);
printf("Resistivity of Dielectric= %.3f*10^12 ohm-m"
   ,row/10^12);
```

Scilab code Exa 9.3 Find max and min electrostatic stresses and capacitance and charging current

```
//Find max and min electrostatic stresses and
   capacitance and charging current
clear;
clc;
//soltion
//given
a=0.645;//cm^2
d=sqrt(4*a/%pi)*0.01;//m//Diameter of conductor
V=11000;//Volts
f=50;//Hz
Er=3.5;//permitivity of the dielectric used
D=0.0218;//m//Internal diameter of sheath
gmax=2*V/(d*log(D/d))/10^5;
printf("Maximum electrostatic stresses= %.2f kV/cm\n
   ",gmax);
gmin=2*V/(D*log(D/d))/10^5;
printf("Minimum electrostatic stresses= %.1f kV/cm\n
   ",gmin);
C=0.024*Er*10^-6/(log10(D/d));
printf("Capacitance of cable= %.2f*10^-6 farad\n",C
   *10^6);
Ic=2*%pi*f*C*V/sqrt(3);
```

```
printf("Charging current per phase per km length= %
    .2f A ",Ic);
```

Scilab code Exa 9.4 Find max electrostatic stresses and charging kVA

```
//Find max electrostatic stresses and charging kVA
clear;
clc;
//soltion
//given
r=0.6;//cm
d=0.025;//m//Diameter of conductor
V=33000;//Volts
f=50;//Hz
l=3.4;//km
Er=3.1;//permitivity of the dielectric used
D=d+2*r*10^-2;//m//Internal diameter of sheath
gmax=2*V/(d*log(D/d));
printf("Maximum electrostatic stresses= %.1f*10^6 V/
   m\n",gmax/10^6);
C=0.024*Er*l*10^-6/(log10(D/d));
printf("Capacitance of cable= %.4f*10^-6 farad\n",C
    *10^6);
Ic=2*%pi*f*C*(V/sqrt(3));
printf("Charging current per phase per km length= %
    .2f A\n",Ic);
kVA=sqrt(3)*V*Ic*10^-3;
printf("Total Charging= %.2f kVAR",kVA);
//THERE ARE SOME CALCULATION ERRORS IN THE BOOK
    BECAUSE OF WHICH Ic=0.3078 A AND TOTAL CHARGING
    CURRENT= 17.57kVAR
```

Scilab code Exa 9.5 Determine internal diameter of shealth D and diameter of conductor d

```
//Determine internal diameter of shealth D and
    diameter of conductor d
clear;
clc;
//soltion
//given
gmax=23*10^5;//V/cm
V=10000;//Volts
d=2*V/gmax;
D=exp(1)*d*1000;
printf("Diameter of conductor(d)= %.1f mm \nInternal
    diameter of shealth (D)= %.2f mm",d*1000,D);
```

Scilab code Exa 9.6 Determine most economical value of diameter and overall diameter of insulation

```
//Determine most economical value of diameter and
    overall diameter of insulation
clear;
clc;
//soltion
//given
gmax=60;//kV/cm
V=132*sqrt(2)/sqrt(3);//kV
d=2*V/gmax;
D=exp(1)*d;
printf("Diameter of conductor(d)= %.1f cm \nInternal
    diameter of shelath= %.2f cm",d,D);
```

Scilab code Exa 9.7 Determine most economical value of diameter of single core cable

```
//Determine most economical value of diameter of
   single core cable
clear;
clc;
//soltion
//given
gmax=40;//kV/cm
V=50*sqrt(2);//kV
d=2*V/gmax;
printf("Diameter of conductor(d)= %.3f cm",d);
```

Scilab code Exa 9.8 find safe working voltage of cable

```
//find safe working voltage of cable
clear;
clc;
//soltion
//given
d=4;//cm
D=10;//cm
e1=5;//realtive permeabilty
e2=4;//realtive permeabilty
e3=3;//realtive permeabilty
d1=e1*d/e2;
d2=e1*d/e3;
gmax=40;//kV/cm
Vper=(gmax/2)*[d*log(d1/d)+d1*log(d2/d1)+d2*log(D/d2
   )];
Vsafe1=Vper/sqrt(2);
printf("Safe working voltage(rms) of a cable= %.2f
   kV\n",Vsafe1);
Vpeak=(gmax/2)*[d*log(D/d)];
```

```
Vsafe2=Vpeak/sqrt(2);
printf("Safe working voltage(rms) of the ungraded cable= %.2f kV",Vsafe2);
```

Scilab code Exa 9.9 find radial thickness and safe working voltage of cable

```
//find radial thickness and safe working voltage of cable
clear;
clc;
//soltion
//given
d=6;//cm
D=18;//cm
e1=5;//realtive permeabilty
e2=4;//realtive permeabilty
g1max=30;//kV/cm
g2max=20;//kV/cm
d1=g1max*e1*d/(g2max*e2);
tin=(d1-d)/2;
printf("Radial thickness of inner dielectric= %.3f cm \n",tin);
tout=(D-d1)/2;
printf("Radial thickness of outer dielectric= %.3f cm \n",tout);
Vper=(g1max/2)*[d*log(d1/d)]+(g2max/2)*(d1*log(D/d1));
Vsafe=Vper/sqrt(2);
printf("Safe working voltage for a cable(rms)= %.2f kV\n",Vsafe);
```

Scilab code Exa 9.10 find the voltage on the intersheaths

```
//find the voltage on the intersheaths
clear;
clc;
//soltion
//given
d=2.5;//cm
d1=3.1;//cm
d2=4.2;//cm
D=6;//cm
V=66*sqrt(2/3);//kV
V1=poly(0,"V1");
V2=poly(0,"V2");
V3=poly(0,"V3");
g1max=V1/((d/2)*log(d1/d));//kV/cm
g2max=V2/((d1/2)*log(d2/d1));//kV/cm
g3max=V3/((d2/2)*log(D/d2));//kV/cm
V2=g1max/2.1244605;
V3=g1max/1.3350825;
V1=roots(V1+V2+V3-V);
V2=V1*1.7542;//after solving g1max=g2max
V3=V1*2.7857;//after solving g1max=g3max
Vf=V-V1;
Vs=V-V1-V2;
printf("Voltage on first intersheath(i.e. near to the core) = %.3f kV\n",Vf);
printf("Voltage on second intersheath= %.3f kV",Vs);
//THERE IS A SLIGHT ERROR DUE TO THE USAGE OF FLOATING POINT
//IN BOOK Vf=44.237 kV & Vs= 27.147kV
```

Scilab code Exa 9.11 find the position and voltage on the intersheaths and max and min stress

```
//find the position and voltage on the intersheaths and max and min stress
```

```
clear;
clc;
//soltion
//given
d=2;//cm
D=5.3;//cm
V=66*sqrt(2/3);//kV
V1=poly(0,"V1");
V2=poly(0,"V2");
V3=poly(0,"V3");
d1=poly(0,"d1")
d1d2=D*d;//d1*d2
d2=(d1^2)/2;
printf("d2= ");
disp(d2);
d1=(2*d1d2)^(1/3);//after putting value of d2 in d1*
   d2
d2=(d1^2)/2;
printf("d1= %.3f cm \nd2= %.1f cm \n",d1,d2);
V2=(d1/d)*V1;
V3=(d2/d)*V1;
V1=roots(V1+V2+V3-V);
V2=(d1/d)*V1;
V3=(d2/d)*V1;
Vf=V-V1;
Vs=V-V1-V2;
printf("Voltage on first intersheath(i.e. near to
   the core) = %.3f kV\n",Vf);
printf("Voltage on second intersheath= %.f kV\n",Vs)
   ;
Gmax=V1/((d/2)*log(d1/d));
Gmin=V1/((d1/2)*log(d1/d));
printf("Maximum stress= %.f kV/cm \nMinimum stress=
   %.2f kV/cm",Gmax,Gmin);
//There is an error in in book it is Voltage on
   second intersheath= 23.91 kV & Gmax and Gmin in
   book it is 39kV/cm & Gmin= 28.35 kV/cm
```

Scilab code Exa 9.12 Calculate the charging current

```
//Calculate the charging current
clear;
clc;
//soltion
//given
C3=(0.4*10^-6)*5;//farad
Vph=11*10^3/sqrt(3);
f=50;//Hz
Cn=2*C3;
Ic=2*%pi*f*Vph*Cn;
printf("Charging current= %.2f A",Ic)
```

Scilab code Exa 9.13 Calculate the kVA taken

```
//Calculate the kVA taken
clear;
clc;
//soltion
//given
C3=(0.2*10^-6)*20;//farad
Vph=11*10^3/sqrt(3);
f=50;//Hz
Cn=2*C3;
Ic=2*%pi*f*Vph*Cn;
printf("Charging current= %.2f A\n",Ic)
kVA=3*Vph*Ic*10^-3;
printf("kVA taken by the cable= %.2f kVA",kVA);
//IN BOOK IT IS 24.75 kVA DUE TO SOME PRINTING
    MISTAKE
```

Scilab code Exa 9.14 Calculate the charging current

```
//Calculate the charging current
clear;
clc;
//soltion
//given
C1=14*10^-6;//farad
C2=8*10^-6;//farad
Ce=C1/3;
Cc=(C2-Ce)/2;
Vph=66*10^3/sqrt(3);
f=50;//Hz
Cn=Ce+3*Cc;
Ic=2*%pi*f*Vph*Cn;
printf("Charging current= %.2f A",Ic);
//In book it is 115.82 A due to some printing
   mistake
```