Scilab Textbook Companion for Theory of Alternating Current Machinery by A. S. Langsdorf[1]

Created by
Ajay Kumar G
Bachelor of Engineering (B.E.)
Electrical Engineering
Sri Jayachamarajendra College of Engineering
College Teacher
R S Ananda Murthy
Cross-Checked by

August 10, 2013

[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: Theory of Alternating Current Machinery

Author: A. S. Langsdorf

Publisher: Tata McGraw - Hill Education

Edition: 1

Year: 1999

ISBN: 978-0-07-099423-2

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

Fundamental Principles of Transformer

Scilab code Exa 1.6.14 To find secondary resistance and reactance

```
// Example1_6_pg14.sce
// To find secondary resistance and reactance
// Theory of Alternating Current Machinery by
   Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 14


clear; clc; close;

// Given data
volt_amp = 10e+3; // Volt Ampere rating of
   transformer is 10kA
volt_ratio = 440/110; // Transformer voltage ratio
freq_tr = 60; // Frequency of transformer usage is
   60cps or 60Hz
pri_res = 0.50; // Primary resistance is 0.50 Ohm
sec_res = 0.032; // Secondary resistance is 0.032
```

```
Ohm
pri_reac = 0.90; // Primary leakage reactance is 0.90 Ohm
sec_reac = 0.06; //Secondary leakage reactance is 0.06 Ohm

// Calculations
printf("The ratio of transformation is %d",volt_ratio);
sec_res_ref_pri = sec_res*(volt_ratio^2); // Ohms
sec_reac_ref_pri = sec_reac*(volt_ratio^2); // Ohms

disp('Hence,');
printf("Secondary resistance referred to the primary = %0.3f Ohm \n",sec_res_ref_pri); // Ohms
printf("Secondary reactance referred to the primary = %0.2f Ohm",sec_reac_ref_pri); // Ohms

// Result
// The ratio of transformation is 4
// Secondary resistance referred to the primary is 0.512 Ohm
// Secondary reactance referred to the primary is 0.96 Ohm
```

Scilab code Exa 1.9.18 To find the secondary terminal voltage

```
// Example1_9_pg18.sce
// To find the secondary terminal voltage
// Theory of Alternating Current Machinery by Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
```

```
// Example in Page 18


clear; clc; close;

// Given data
v1 = 2000; // Primary voltage , volts
v2 = 400; // Secondary Open Voltage , volts
pf = +0.8; // Power factor lagging 80%
r1 = 5.5; // Resistance R1, Ohms
r2 = 0.2; // Resistance R2, Ohms
x1 = 12; // Reactance X1, Ohms
x2 = 0.45; // Reactance X2, Ohms
va_rating = 10e+3 // volt-ampere rating of
   transformer , VA
voltage1 = v1; // Supply input voltage , Volts

// Calculations
current1 = va_rating/voltage1; // Amperes
current2 = current1; // Amperes
turns_ratio = v1/v2;
r2dash = turns_ratio^2 * r2; // r2 as referred to
   primary side , Ohms
sum_ofr = r1 + r2dash; // total equivalent
   resistance referred to primary , Ohms
x2dash = turns_ratio^2 * x2; // x2 as referred to
   primary side , Ohms
sum_ofx = x1 + x2dash; // Sum of reactances , Ohms
// Taking current axis as the reference as per the
   problem
vec_current1 = 5 + 0*%i; // Vector Current 1,
   Amperes
vec_current2 = vec_current1; // Vector Current 2,
   Amperes
theta = acos(0.8); // lagging phase angle in radians
vector_volt1 = voltage1; // Volts
function y = ff(voltage2)
  // To solve for secondary voltage from the
```

```
equation
  //      vector_volt1 = vector_volt2 + vec_current2*((sum_ofr)+(sum_ofx)*%i);
  //      vector_volt2 = voltage2*(cos(theta)+sin(theta)*%i);
  //      vector_volt1 = voltage2*(cos(theta)+sin(theta)*%i) + vec_current2*((sum_ofr)+(sum_ofx)*%i);
  // Separating real and imaginary parts and calculating the absolute values, and equating it to zero(or here y(1)), the expression would look like below
  //  y(1) = -(vector_volt1^2) + (cos(theta)*voltage2(1) + abs(vec_current2)*sum_ofr)^2 + (voltage2(1)*sin(theta) + abs(vec_current2)*sum_ofx)^2;
  y(1) = -(vector_volt1^2) + (cos(theta)*voltage2(1) + abs(vec_current2)*(sum_ofr))^2 + (sin(theta)*voltage2(1) + abs(vec_current2)*(sum_ofx))^2;
  endfunction
sec_volt_in_terms_of_pri = fsolve ([0.1], ff); // in Volts
sec_voltage = sec_volt_in_terms_of_pri/turns_ratio; // in Volts
printf("\nSecondary Voltage as referred to primary is %.2f volts \n", sec_volt_in_terms_of_pri);
printf("Secondary Terminal Voltage at full load is %.2f volts \n", sec_voltage);


// Result
// Secondary Voltage as referred to primary is 1887.30 volts
// Secondary Terminal Voltage at full load is 377.46 volts
```

Scilab code Exa 1.13.28 To find the regulation of transformer

```
// Example1_13_pg28.sce
// To find the regulation of transformer
// Theory of Alternating Current Machinery by Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 28


clear; clc; close;

// Given data
v1 = 1100; // Primary voltage, Volts
v2 = 110; // Secondary Open Voltage, Volts
volt_sc = 33; // Voltage for Short Circuit full load current, Volts
pow_sc_in = 85; // Short Circuit input Power, Watts
pf = +0.8; // Power factor lagging 80%
va_rating = 5e+3 // volt-ampere rating of transformer, VA

// Calculations

// Method based on Eq. 1-35
// v1^2 = (v2 + volt_sc*cos(thetae - theta2))^2 + (volt_sc*sin(thetae - theta2))^2;
current1 = va_rating/v1; // Current in Amperes
thetae = acos(pow_sc_in /( volt_sc * current1 ));
theta2 = acos(pf);
function y = ff1(v2)
```

```
  y(1) = -(v1^2) + (v2 + volt_sc*cos(thetae - theta2))^2 + (volt_sc*sin(thetae - theta2))^2;
  endfunction
volt2 = fsolve ([0.1], ff1); // voltage in volts
// Regulation = ( (v1 - volt2)/v1 ) *100
Regulation1 = ((v1 - volt2)/v1)*100;
printf("\nRegulation of the Transformer by method 1 is %.2f %% \n", Regulation1);

// Method based on Eq. 1-36
// v1^2 = (v2 + current1*re*cos(theta2) + current1*xe*sin(theta2))^2 + (current1*xe*cos(theta2) - current1*re*sin(theta2))^2;
current1 = va_rating/v1; // Current in Amperes
thetae = acos(pow_sc_in /( volt_sc * current1 ));
theta2 = acos(pf);
ze = volt_sc/current1; // impedance in Ohms
re = pow_sc_in/(current1^2); // Resistance in Ohms
xe = (ze^2 - re^2)^0.5; // Reactance in Ohms
function y = ff2(v2)
  y(1) = -(v1^2) + (v2 + current1*re*cos(theta2) + current1*xe*sin(theta2))^2 + (current1*xe*cos(theta2) - current1*re*sin(theta2))^2;
  endfunction
volt2 = fsolve ([0.1], ff2);
// Regulation = ( (v1 - volt2)/v1 ) *100
Regulation2 = ((v1 - volt2)/v1)*100;
printf("Regulation of the Transformer by method 2 is %.2f %% \n", Regulation2);

// Result
// Regulation of the Transformer by method 1 is 2.85 %
// Regulation of the Transformer by method 2 is 2.85 %
```

Scilab code Exa 1.14.29 To find regulation by percent method

```
// Example1_14_pg29.sce
// To find regulation by percent method
// Theory of Alternating Current Machinery by Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 29


clear; clc; close;

// Given data
v1 = 1100; // Primary voltage, volts
v2 = 110; // Secondary Open Voltage, volts
volt_sc = 33; // Voltage for Short Circuit full load current, volts
pow_sc_in = 85; // Short Circuit input Power, watts
pf = +0.8; // Power factor lagging 80%
va_rating = 5e+3 // volt-ampere rating of transformer, VA

// Calculations

// Method based on Eq. 1-38
// %regulation = rpc*cos(theta2) + xpc*sin(theta2) + ((xpc*cos(theta2) - rpc*sin(theta2))^2)/200;
current1 = va_rating/v1; // Current in Amperes
thetae = acos(pow_sc_in /( volt_sc * current1 ));
theta2 = acos(pf);
ze = volt_sc/current1; // Impedance in Ohms
```

```
re = pow_sc_in/(current1^2); // Resistance in Ohms
xe = (ze^2 - re^2)^0.5; // Impedance in Ohms
rpc = (current1*re/v1)*100;
xpc = (current1*xe/v1)*100;
percent_regulation = rpc*cos(theta2) + xpc*sin(
    theta2) + ((xpc*cos(theta2) - rpc*sin(theta2))^2)
    /200;
printf("Regulation of the Transformer by per-cent
    method is %.2f %% \n", percent_regulation);

// Result
// Regulation of the Transformer by per-cent method
    is 2.85 %
```

Scilab code Exa 1.14.31 To find the per unit regulation

```
// Example1_14_pg31.sce
// To find the per unit regulation
// Theory of Alternating Current Machinery by
    Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 31


clear; clc; close;

// Given data
r_pu = 0.017; // Per-unit resistance
x_pu = 0.0247; // Per-unit reactance
power_factor = 1; // Unity Power Factor
overload = 0.25; // 25% overload

```

```
// Calculations
phi = acos(power_factor);
OL_factor = 1.00 + overload;
r_pu = r_pu*OL_factor; // Base value has to be
    changed for 0.25 overload
x_pu = x_pu*OL_factor; // Base value has to be
    changed for 0.25 overload
// Formula for regulation is , Per-unit-regulation =
    r_pu*cos(phi) + x_pu*sin(phi) + 0.5*(x_pu*cos(phi
    ) - r_pu*sin(phi))^2
perunit_regulation = r_pu*cos(phi) + x_pu*sin(phi) +
     0.5*(x_pu*cos(phi) - r_pu*sin(phi))^2;

// disp('Hence,');
printf("Per-unit regulation = %0.4f",
    perunit_regulation);

// Result
// Per-unit regulation = 0.0217
```

Scilab code Exa 1.14.33 To find the load loss of transformer

```
// Example1_15_pg33.sce
// To find the load loss of transformer
// Theory of Alternating Current Machinery by
    Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 33


clear; clc; close;

```

```
// Given data
Total_Culoss1 = 630; // Total Copper Loss at 20 degree celcius, watts
TrueCopper_loss1 = 504; // Copper loss due to True Ohmic resistance at 20degree celcius, watts
temp1 = 20; // Temperature, degree celcius
temp2 = 75; // Temperature, degree celcius

// Calculations
eddy_loss1 = Total_Culoss1 - TrueCopper_loss1; // Eddy Current loss at 20 degree celsius, watts
TrueCopper_loss2 = TrueCopper_loss1 * (temp2 + 234.5) / (temp1 + 234.5); // True Copper loss at 75 degree celcius, watts
eddy_loss2 = eddy_loss1 * (temp1 + 234.5) / (temp2 + 234.5);// Eddy Current loss at 75 degree celsius, watts
load_loss = TrueCopper_loss2 + eddy_loss2; // Load loss at 75 degree celsius, watts

printf("Eddy Current loss at 20 degree celcius = %.0f watts\n", eddy_loss1);
printf("True Copper loss at 75 degree celcius = %.0f watts\n", TrueCopper_loss2);
printf("Load loss at 75 degree celcius = %.0f watts", load_loss);

// Result
// Eddy Current loss at 20 degree celcius = 126 watts
// True Copper loss at 75 degree celcius = 613 watts
// Load loss at 75 degree celcius = 717 watts
```

Scilab code Exa 1.16.37 To measure the core loss of transformer

```
// Example1_16_pg37.sce
// To measure the core loss of transformer
// Theory of Alternating Current Machinery by
   Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 37


clear; clc; close;

// Given data
f1 = 30; // Frequency, Hz
B1 = 8; // Flux Density, kilogauss
P1 = 0.135; // Core loss, watts per lb
f2 = 60; // Frequency, Hz
B2 = 12; // Flux Density, kilogauss
P2 = 0.75; // Core loss, watts per lb
P3 = 0.31; // Core loss, watts per lb

// Calculations
a = f2/f1;
x=(log(B2^2*(P2 - a^2 * P3)/((P2 - a*P3)*B1^2 - a*(a
   -1)*P1*B2^2)))/(log(B2/B1));
kh = (P2 - a^2 * P3)/(f2*(1 - a )*(B2^x));
ke = ((P2 - a*P3)*a)/((a-1)*f2^2*B2^2);
Ph1 = kh*f1*B1^x;  Pe1 = ke*f1^2*B1^2; // Hysteresis
    Power loss, watts
Ph2 = kh*f2*B2^x;  Pe2 = ke*f2^2*B2^2; // Hysteresis
    Power loss, watts
Ph3 = kh*f1*B2^x;  Pe3 = ke*f1^2*B2^2; // Hysteresis
    Power loss, watts
Pt1 = Ph1 + Pe1; // Total Power loss, watts
Pt2 = Ph2 + Pe2; // Total Power loss, watts
Pt3 = Ph3 + Pe3; // Total Power loss, watts
disp('Value of x is'); disp(x);
```

```
disp('Value of kh is'); disp(kh);
disp('Value of ke is'); disp(ke);

printf("\n
    ---------------------------------------------------------------------------\
    n    f    | B,kilogauss  | Ph,watts per lb  | Pe,watts
     per lb    \n
    ---------------------------------------------------------------------------\
    n    %d  |         %d        |         %.3f         |         %.3
    f          \n    %d  |         %d        |         %.3f         |
          %.3f          \n    %d  |         %d        |         %.3f
          |          %.3f          \n
    ---------------------------------------------------------------------------\
    n", f1, B1, Ph1, Pe1, f2, B2, Ph2, Pe2, f1, B2,
    Ph3, Pe3);

// Result
//
// Value of x is
//
//     2.0637323
//
// Value of kh is
//
//     0.0000484
//
// Value of ke is
//
//     0.0000005
//
//
    ---------------------------------------------------------------------------

//    f    | B,kilogauss  | Ph,watts per lb  | Pe,watts
    per lb
//
    ---------------------------------------------------------------------------
```

```
//     30  |          8        |         0.106          |          0.029
//     60  |         12        |         0.490          |          0.260
//     30  |         12        |         0.245          |          0.065
//
   ____________________________________________________________
```

Scilab code Exa 1.17.41 To find the efficiency at different loads

```
// Example1_17_pg41.sce
// To find the efficiency at different loads
// Theory of Alternating Current Machinery by
   Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 41


clear; clc; close;

// Given data
va = 50e+3; // VA rating of transformer, VA
v1 = 2200; // Volts
v2 = 220; // Volts
f = 60; // Frequency, Hz
core_loss = 350; // Power loss, watts
cu_loss = 630; // Power loss, watts
pf0 = 1;
pf1 = 0.8;

// Calculations
turns_ratio = v1/v2;
upf_full_load_eff = (va*pf0/(va*pf0 + core_loss +
```

```
    cu_loss))*100; // Full Load Efficiency at upf
upf_three_fourth_eff = ((0.75*va*pf0)/(0.75*va*pf0 +
     core_loss + (0.75^2)*cu_loss))*100; //
    Efficiency at three-fourth load at upf
full_load_eff = ((va*pf1)/(va*pf1 + core_loss +
    cu_loss))*100; // Efficiency at full load at 0.8
    pf
three_fourth_eff = ((0.75*va*pf1)/(0.75*va*pf1 +
    core_loss + (0.75^2)*cu_loss))*100; // Efficiency
     at three-fourth load at 0.8pf

printf('Efficiency at Full load & unity power factor
     = %.1f %% \n ',upf_full_load_eff);
printf('Efficiency at Three-fourth the full load &
    unity power factor = %.1f %%\n ',
    upf_three_fourth_eff);
printf('Efficiency at Full load efficiency at 80%%
    power factor = %.1f %%\n ',full_load_eff);
printf('Efficiency at three-fourth load efficiency
    at 80%% power factor = %.1f %%\n ',
    three_fourth_eff);

// Result
// Efficiency at Full load & unity power factor =
    98.1 %
// Efficiency at Three-fourth the full load & unity
    power factor = 98.2 %
// Efficiency at Full load efficiency at 80% power
    factor = 97.6 %
// Efficiency at three-fourth load efficiency at 80%
     power factor = 97.7 %
```

Chapter 2

Transformer Connections and Operation

Scilab code Exa 2.3.69 To find primary voltage and current supplied

```
// Example2_3_pg69.sce
// To find primary voltage and current supplied
// Theory of Alternating Current Machinery by
   Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 69


clear; clc; close;

// Given data

// Transformer A data
va_A = 100e+3; // VA rating of Transformer
v1_A = 4600; // Voltage in volts
v2_A = 230; // Voltage in volts
x_A = 0.027; // Reactance in Ohms
r_A = 0.008; // Resistance in Ohms
```

```

// Transformer B data
va_B = 200e+3; // VA rating of Transformer
v1_B = 4610; // Voltage in volts
v2_B = 225; // Voltage in volts
x_B = 0.013; // Reactance in ohms
r_B = 0.003; // Resistance in ohms

// Common Data
P_load = 150e+3; // Power in Watts
pf = +0.85; // + denotes lagging power factor
vg = 225; // Voltage in volts


// Calculations

// Transformer A
a_1 = v1_A / v2_A;
z_1 = r_A + x_A*%i;
y_1 = 1 / z_1;
y_1_HVside = y_1 / a_1;

// Transformer B
a_2 = v1_B / v2_B;
z_2 = r_B + x_B*%i;
y_2 = 1 / z_2;
y_2_HVside = y_2 / a_2;

y_K = y_1 + y_2;
y_K_HVside = y_1_HVside + y_2_HVside;

// To find the current
I = P_load / (vg * pf) ;
V2_vec = vg;
theta = acos(0.85);
I_vec = I*(cos(theta) - sin(theta)*%i); // - sign indicates I lags V

```

```
V1_vec = ((V2_vec * y_K) + I_vec) / (y_K_HVside) ;

I1_vec = (I_vec + V1_vec*((y_K / a_1) - y_K_HVside))
    / (z_1 * y_K );

I2_vec = I_vec - I1_vec;

printf(' Primary Voltage of transformer = %f /_ %f
   Volts\n', abs(V1_vec), (atan((imag(V1_vec))/(real
   (V1_vec))))*180/%pi);
printf(' Current Supplied by transformer A = %f /_
   %f Volts\n', abs(I1_vec), (atan((imag(I1_vec))/(
   real(I1_vec))))*180/%pi);
printf(' Current Supplied by transformer B = %f /_
   %f Volts\n', abs(I2_vec), (atan((imag(I2_vec))/(
   real(I2_vec))))*180/%pi);

// Result
// Primary Voltage of transformer = 4678.867698 /_
   1.211839 Volts
// Current Supplied by transformer A = 361.324403 /_
    -44.400715 Volts
// Current Supplied by transformer B = 438.858386 /_
    -21.431553 Volts
```

Scilab code Exa 2.6.76 To find branch currents and voltages

```
// Example2_6_pg76.sce
// To find branch currents and voltages
// Theory of Alternating Current Machinery by
   Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
```

```
// Example in Page 76


clear; clc; close;

// Given data

// Transformer data
va = 100e+3; // VA rating of Transformer
v1 = 11500; // Voltage in volts
v2 = 230; // Voltage in volts
f = 60; // Frequency in Hz
OC_pow = 560; // Power in watts
pf = +0.155;
sc_volt = 217.5; // Volts
sc_curr = 8.7; // Amperes
sc_pow = 1135; // Power in watts
ll_volt = 15000; // Line to line voltage
z_1 = 0.6; // Impedance
pf2 = +0.866;
pf3 = -0.5;

// Calculations

power_factor = sc_pow / (sc_volt * sc_curr) ;
theta_e = acos(power_factor);
transformation_ratio = v1 / v2 ;

// HT values

z = sc_volt / sc_curr;
r = z*cos(theta_e);
x = z*sin(theta_e);

// LT values

z_lt = z/(transformation_ratio^2) ;
r_lt = r/(transformation_ratio^2) ;
```

```
x_lt = x/(transformation_ratio^2) ;

zz = r_lt + %i*x_lt ;

// Referring to figure 2.16(b) in page 77

z1 = z_1 + zz ;
z_2 = z_1*(pf2 + %i*abs(pf3));
z2 = z_2 + zz;
z_3 = z_1*(abs(pf3) - %i*pf2);
z3 = z_3 + zz;

disp('z1 = ')
disp(z1);

disp('z2 = ')
disp(z2);

disp('z3 = ')
disp(z3);

disp('By referring to Figure 2.16(b) in page 77, E_A
    , E_B, E_C can be written in terms of the
    unknowns x and y.');

printf("\nE_A = -(x - 150) + j(259.8 - y) \nE_B = -x
     - jy \nE_C = (300 - x) - jy");
printf("\n\nI_A = E_A / z1 \nI_B = E_B / z2 \nI_C =
    E_C / z3 \n");

printf("\nI_A = -1.649x -0.0218y +253.01 + j(425.14
    -1.649y +0.0218x) \nI_B = -1.415x -0.829y + j
    (0.829x - 1.415y) \nI_C = -0.860x +1.439y +258 +
    j(-1.439x -0.860y +431.7)\n");

// I_A + I_B + I_C = 0;

disp('On simplification and by separating the real
```

```
    and imaginary parts, we get two equations
    consisting of x and y as variables as shown');

printf("\n -3.924x +0.588y +511.01 = 0\n -0.588x
    -3.924y +856.84 = 0\n");

function y = ff(x);
  y(1) = -3.924*x(1)+0.588*x(2)+511.01;
  y(2) = -0.588*x(1)-3.924*x(2)+856.84;
endfunction
answer = fsolve([100;100],ff);

// Answers given in prob is supposed to have some
    mistake in values of x and y

x = answer([1]);
y = answer([2]);

E_A = -(x - 150) + %i*(259.8 - y) ;
E_B = -x - %i*y ;
E_C = (300 - x) - %i*y;

I_A = E_A / z1 ;
I_B = E_B / z2 ;
I_C = E_C / z3 ;

printf("\n\nI_A = %0.2f /_ %0.2f Amps", abs(I_A),
    atan(imag(I_A)/real(I_A))*180/%pi);
printf("\n\nI_B = %0.2f /_ %0.2f Amps", abs(I_B),
    atan(imag(I_B)/real(I_B))*180/%pi);
printf("\n\nI_C = %0.2f /_ %0.2f Amps", abs(I_C),
    atan(imag(I_C)/real(I_C))*180/%pi);
printf("\n\nE_A = %0.2f /_ %0.2f Volts", abs(E_A),
    atan(imag(E_A)/real(E_A))*180/%pi);
printf("\n\nE_B = %0.2f /_ %0.2f Volts", abs(E_B),
    atan(imag(E_B)/real(E_B))*180/%pi);
printf("\n\nE_C = %0.2f /_ %0.2f Volts", abs(E_C),
    atan(imag(E_C)/real(E_C))*180/%pi);
```

```

// Result
// z1 =
//
//    0.6059982 + 0.0080014i
//
// z2 =
//
//    0.5255982 + 0.3080014i
//
// z3 =
//
//    0.3059982 - 0.5115986i
//
// By referring to Figure 2.16(b) in page 77, E_A,
   E_B, E_C can be written in terms of the unknowns
   x and y.
// E_A = -(x - 150) + j(259.8 - y)
// E_B = -x - jy
// E_C = (300 - x) - jy
//
// I_A = E_A / z1
// I_B = E_B / z2
// I_C = E_C / z3
//
// I_A = -1.649x -0.0218y +253.01 + j(425.14 -1.649y
    +0.0218x)
// I_B = -1.415x -0.829y + j(0.829x - 1.415y)
// I_C = -0.860x +1.439y +258 + j(-1.439x -0.860y
   +431.7)
//
// On simplification and by separating the real and
   imaginary parts, we get two equations consisting
   of x and y as variables as shown
//
// -3.924x +0.588y +511.01 = 0
// -0.588x -3.924y +856.84 = 0
//
```

```
//
// I_A = 108.89 /_ -82.59 Amps
//
// I_B = 412.73 /_ 20.30 Amps
//
// I_C = 402.59 /_ 4.99 Amps
//
// E_A = 65.99 /_ -81.84 Volts
//
// E_B = 251.44 /_ 50.67 Volts
//
// E_C = 240.00 /_ -54.13 Volts
```

Scilab code Exa 2.22.111 Conductively and Inductively transferred power

```
// Example2_22_pg111.sce
// Conductively and Inductively transferred power
// Theory of Alternating Current Machinery by
    Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 111


clear; clc; close;

// Given data

// Transformer data
va = 10e+3; // VA rating of Transformer, VA
v1 = 2300; // Voltage in volts
v2 = 230; // Voltage in volts
disp('Referring to Fig 2.57, we have');
```

```
// Calculations
V_1 = v1 + v2; // Voltage in volts
I_1 = va/v2; // Voltage in volts
I_3 = va/v1; // Voltage in volts
I_2 = I_1 + I_3; // Current in Amperes
a = V_1 / v1;
P = V_1 * I_1; // Power in watts
P_i = P * (a - 1)/a; // Power in watts
P_c = round(P/a); // Power in watts

printf("\n\nTotal volt-amperes supplied from the
    source is = %d VA \nVolt-Amperes supplied
    inductively is = %d VA\nPower supplied
    conductively is %d VA\n", P, P_i, P_c);

// Result
//  Referring to Fig 2.57, we have
//
//
// Total volt-amperes supplied from the source is =
    110000 VA
// Volt-Amperes supplied inductively is = 10000 VA
// Power supplied conductively is 100000 VA
```

Scilab code Exa 2.29.130 Positive and negative sequence voltages

```
// Example2_29_pg130.sce
// Positive and negative sequence voltages
// Theory of Alternating Current Machinery by
    Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
```

```
// Example in Page 130


clear; clc; close;

// Given data

V_1 = 1000 + %i*50;
V_2 = -800 + %i*100;
V_3 = -200 - %i*150;
a = cos(2*%pi/3) + %i*sin(2*%pi/3);

// Calculations

disp('According to Equations 2-88 and 2-89 in page 130');
V_1p = (V_1 + V_2*a + V_3*a^2)/3;
V_1n = (V_1 + V_2*a^(-1) + V_3*a^(-2))/3;

printf("\n\nPositive sequence voltage is = %0.4f /_ %0.2f Volts \nNegative sequence voltage is = %0.4f /_ %0.2f Volts\n", abs(V_1p),atan(imag(V_1p)/real(V_1p))*180/%pi, abs(V_1n),atan(imag(V_1n)/real(V_1n))*180/%pi);

// Result
//  According to Equations 2-88 and 2-89 in page 130
//
//
// Positive sequence voltage is = 452.7740 /_ -19.11 Volts
// Negative sequence voltage is = 605.5265 /_ 19.11 Volts
```

Scilab code Exa 2.29.131 Positive Negative and Zero sequence voltages

```
// Example2_29_pg131.sce
// Positive Negative and Zero sequence voltages
// Theory of Alternating Current Machinery by Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 131


clear; clc; close;

// Given data
V_1 = 1000 + 50*%i;
V_2 = -800 + 100*%i;
V_3 = -1100 - 270*%i;
a = cos(2*%pi/3) + %i*sin(2*%pi/3);

// Calculations
disp('According to Equations 2-90, 2-88 and 2-89');
V_0 = (V_1 + V_2 + V_3)/3;
V_1p = (V_1 + V_2*a + V_3*a^2)/3;
V_1n = (V_1 + V_2*a^(-1) + V_3*a^(-2))/3;

printf("\n\nZero sequence voltage is = %0.4f /_ %0.2f Volts \nPositive sequence voltage is = %0.4f /_ %0.2f Volts \nNegative sequence voltage is = %0.4f /_ %0.2f Volts\n",abs(V_0),atan(imag(V_0)/real(V_0))*180/%pi, abs(V_1p),atan(imag(V_1p)/real(V_1p))*180/%pi, abs(V_1n),atan(imag(V_1n)/real(V_1n))*180/%pi);

// Result
//  According to Equations 2-90, 2-88 and 2-89
//
//
// Zero sequence voltage is = 302.6549 /_ 7.59 Volts
```

```
// Positive sequence voltage is = 558.9050 /_ 13.62 Volts
// Negative sequence voltage is = 757.9524 /_ -3.15 Volts
```

Chapter 3

Transformer structure Insulation Heating and Load Stresses

Scilab code Exa 3.16.161 To find radial force due to current

```
// Example3_16_pg161.sce
// To find radial force due to current
// Theory of Alternating Current Machinery by
   Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 161


clear; clc; close;

// Given data
va = 200e+3;  // Volt Amperes of transformer, VA
v1 = 11000; // Voltage in volts
v2 = 2300; // Voltage in volts
T = 46.3; // Mean length of the turn, inches
n = 455; // Number of turns
```

```
I = 1320; // Current in Amperes
l = 35; // length in inches
k = 1.8;
zeq_ht = 8.33;

// Calculations

F_av = (0.45/1e+7)*((T*n^2*I^2)/(k*l));
printf("\n The radial force due to the current of %d
    Amps for given data is %d lb\n", I, round(F_av))
    ;

// Result
//  The radial force due to the current of 1320 Amps
    for given data is 11930 lb
```

Chapter 10

The Synchronous Generator

Scilab code Exa 10.9.407 To find the field excitation required

```
// Example10_9_pg407.sce
// To find the field excitation required
// Theory of Alternating Current Machinery by
   Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 407


clear; clc; close;

// Given data
va = 2500e+3; // Volt Ampere rating of machine, VA
vll = 6600; // Line to Line voltage in volts
N = 3000; // Number of turns
f = 50; // Frequency in Hz
slots = 60;
n = 4;
poles =2;
r = 0.073;
x = 0.87;
```

```
pf1 = 0.8;
pf2 = 1;
pf3 = 0;
phase = 3;

// Calculations

// For 80% power factor

phi = acos(pf1);
V = vll / sqrt(3);
I = round(va / (phase*V)) ;
IR_a = I*r;
IX_a = I*x;
V_vec = V*(cos(phi) +%i*sin(phi));
E = V_vec + I*(r + %i*x);
E_mag = sqrt(real(E)^2 + imag(E)^2);
conductors = slots * n;
turns = conductors/2;
N_p = turns / (poles * phase);
q = slots / (poles * phase);
gama = 360 / slots;
gama = gama*%pi/2;
k_b1 = (sin(q*gama/2))/(q*sin(gama/2));
k_p1 = 1;
A = (2*sqrt(2)/%pi)*phase*k_b1*k_p1*N_p*I;
cos_alpha = (real(E)/E_mag);
sin_alpha = (imag(E)/E_mag);
alpha = acos(cos_alpha);
F_r_mag = 17500;
F_r = F_r_mag*(cos(alpha + %pi/2) + %i*sin(alpha +
    %pi/2));
F = F_r - A;
F_mag = sqrt(real(F)^2 + imag(F)^2);
disp('The open-circuit voltage corresponding to this
     excitation, determined from Fig. 10-12, is 4450
    volts;');
oc_volt = 4450;
```

```
regulation80 = ((oc_volt - V)/V)*100;
printf("\n\nThe regulation for 80%% power factor is
    %0.1f %% ", regulation80);

// For power factor 1.0

phi = acos(pf2);
V_vec = V*(cos(phi) +%i*sin(phi));
E = V_vec + I*(r + %i*x);
E_mag = sqrt(real(E)^2 + imag(E)^2);
cos_alpha = (real(E)/E_mag);
sin_alpha = (imag(E)/E_mag);
alpha = acos(cos_alpha);
F_r_mag = 16500;
F_r = F_r_mag*(cos(alpha + %pi/2) + %i*sin(alpha +
    %pi/2));
F = F_r - A;
F_mag = sqrt(real(F)^2 + imag(F)^2);
disp('The open-circuit voltage corresponding to this
     excitation, determined from Fig. 10-12, is 4150
    volts;');
oc_volt = 4150;
regulation100 = ((oc_volt - V)/V)*100;
printf("\n\nThe regulation for 100%% power factor is
     %0.1f %% ", regulation100);

// For power factor 0

phi = acos(pf3);
E = V + I*(x);
F_r_mag = 18000;
F_r = F_r_mag + 11300;
printf("\nThe value F_R corresponding to Fig 10-12
    is %d Volts\n", F_r);
disp('The open-circuit voltage corresponding to this
     excitation, determined from Fig. 10-12, is 4500
    volts;');
oc_volt = 4500;
```

```
regulation0 = ((oc_volt - V)/V)*100;
printf("\nThe regulation for 0%% power factor is %0
    .1f %% \n", regulation0);

// Result
//  The open-circuit voltage corresponding to this
    excitation, determined from Fig. 10-12, is 4450
    volts;
//
// The regulation for 80% power factor is 16.8 %
// The open-circuit voltage corresponding to this
    excitation, determined from Fig. 10-12, is 4150
    volts;
//
// The regulation for 100% power factor is 8.9 %
// The value F_R corresponding to Fig 10-12 is 29300
     Volts
//
// The open-circuit voltage corresponding to this
    excitation, determined from Fig. 10-12, is 4500
    volts;
//
// The regulation for 0% power factor is 18.1 %
```

Scilab code Exa 10.10.413 Regulation by emf method

```
// Example10_10_pg413.sce
// Regulation by emf method
// Theory of Alternating Current Machinery by
    Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 413
```

```
clear; clc; close;

// Given data
va = 2500e+3; //Volt-Ampere rating of the
    transformer, VA
vll = 6600; // Line to Line voltage in volts
r = 0.073; // Resistance in Ohms
pf1 = 0.8;
phase = 3;
vref = 3640; // Reference for voltage in volts
iref = 340; // Reference for current in Amperes

// Calculations
z_s = vref/iref;
x_s = sqrt(z_s^2 - r^2);
disp('By Referring to Fig. 10-19');
phi = acos(pf1);
V = vll / sqrt(3);
I = round(va / (phase*V)) ;
V_vec = V*(cos(phi) +%i*sin(phi));
E = V_vec + I*(r + %i*x_s);
E_mag = sqrt(real(E)^2 + imag(E)^2);
Regulation = ((E_mag - V)/V)*100;

printf(" Regulation is found to be %.2f %%",
    Regulation);



// Result
// By Referring to Fig. 10-19
// Regulation is found to be 45.73 %
```

Scilab code Exa 10.12.416 Regulation by mmf method

```
// Example10_12_pg416.sce
// Regulation by mmf method
// Theory of Alternating Current Machinery by
    Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 416

clear; clc; close;

// Given data
va = 2500e+3; // Volt Ampere rating of the
    transformer, VA
vll = 6600; // Line to Line voltage, Volts
r = 0.073; // Resistance in Ohms
x = 0.87; // Reactance in Ohms
pf1 = 0.8;
phase = 3;

// Calculations

phi = acos(pf1);
V = vll / sqrt(3);
I = round(va / (phase*V)) ;
IR_a = I*r;
IX_a = I*x;
V_vec = V*(cos(phi) +%i*sin(phi));
E = V_vec + IR_a;
E_mag = sqrt(real(E)^2 + imag(E)^2);
F_r1_mag = 16500;
```

```
cos_alpha = (real(E)/E_mag);
sin_alpha = (imag(E)/E_mag);
alpha = acos(cos_alpha);
F_r1 = F_r1_mag*(cos(%pi/2 + alpha) + %i*sin(%pi/2 + alpha));
A_plus_Ax = 10000;
F = F_r1 - (A_plus_Ax);
F_mag = sqrt(real(F)^2 + imag(F)^2);
printf("\n Magnitude of F is %0.2f amp-turns per pole", F_mag);
disp('This magnitude of F corresponds to Open-circuit voltage of 4330 Volts');
oc_volt = 4330;
regulation = ((oc_volt - V)/V)*100;
printf("\nRegulation is found to be %0.1f %% \n", regulation);

// Result
//  Magnitude of F is 23866.02 amp-turns per pole
// This magnitude of F corresponds to Open-circuit voltage of 4330 Volts
//
// Regulation is found to be 13.6 %
```

Chapter 16

The Mercury Arc Rectifier

Scilab code Exa 16.9.617 Effect of phase control

```
// Example16_9_pg617.sce
// Effect of phase control
// Theory of Alternating Current Machinery by
   Alexander Langsdorf
// First Edition 1999, Thirty Second reprint
// Tata McGraw Hill Publishing Company
// Example in Page 617


clear; clc; close;

// Given data

phi = 20;
alpha1 = 30;
alpha2 = 0;

// Calculations

ans1 = (cos(phi*%pi/(180*2))*cos(phi*%pi/(180*2) +
   alpha1*%pi/180)*100);
```

```
ans2 = round(cos(phi*%pi/(180*2))*cos(phi*%pi
   /(180*2) + alpha2*%pi/180)*100);
Effect = (ans1/ans2)*100;

printf("\n\nEffect of phase control here is to
   reduce the dc voltage to %0.2f %% of the value it
    would have in the absence of phase control\n",
   Effect);

// Result
// Effect of phase control here is to reduce the dc
   voltage to 77.77 % of the value it would have in
   the absence of phase control
```