เอกสารอ้างอิง

1. Neagle N.M. and Samson D.R., "Loss Reduction from Capacitors Installed on Distribution Feeders", AIEE Transactions, Vol. 75, part III, pp. 950-959, 1956.
2. Cook R.F., "Analysis of Capacitors Application as Affected by Load Cycle", AIEE Transactions, Vol.78, part III, pp.950-957, 1959.
3. Maxwell M., "The Economic Application of Capacitors to Distribution Feeders", AIEE Transactions, Vol.79, pp.353-359, 1960.
4. Cook R.F., "Optimizing the Application of Shunt Capacitors for Reactive-Volt-Ampare Control and Loss Reduction", AIEE Transactions, Vol.83, pp.430-444, August 1961.
5. Cook R.F., "Calculating Loss Reduction Afforded by Shunt Capacitor Application", IEEE Transactions on Power Apparatus and Systems, Vol.83, pp.1227-1230, 1964.
6. Schmill J.V., "Optimum Size and Location of Shunt Capacitors on Distribution Feeders", IEEE Transactions on Power Apparatus and Systems, Vol.87, pp.825-833, September 1968.
7. Duran H., "Optimum number, Location and Size of Shunt Capacitors in Radial Distribution Feeders", IEEE Transactions on Power Apparatus and Systems, Vol.87, pp.1769-1774, October 1969.
8. Chang N.E., "Determination of Primary Feeder Losses", IEEE Transactions on Powe Apparatus and Systems, Vol.88, pp.1574-1577, October 1969.
9. Chang N.E., "Location Shunt Capacitors on Primary Feeder for Voltage Control and Loss Reduction", IEEE Transactions on Power Apparatus and Systems, Vol.91, pp.2189-2195, 1972.
10. Rankine L.J., "Method of Location Shunt Capacitors Suitable for

Computer Solution", Electrical World, September 26, pp.72-73, 1960.

11. Shuford G.E. and Bryan J., "Unique Computer Optimization Program Obsoletes Rule of Thumb Distribution System Improvement Methods", Proceedings of the American Power Conference, April 1977.

12. Hopkinson R.H., "Capacitor Application as an Economic Tool", (Private Correspondence).

13. Grainger J.J., Lee S.H., Byrd A.M. and Clinard K.N., "Proper Placement of Capacitors for Losses Reduction on Distribution Primary Feeders", Proceeding of the American Power Conference, Vol.42, pp.593-603, 1980.

14. Lee S.H. and Grainger J.J., "Optimum Placement of Fixed and Switched Capacitors on Primary Distribution Feeders", IEEE Transaction Power Apparatus and Systems, Vol.100, pp.345-351, January 1981.

15. Grainger J.J. and Lee S.H., "Optimum Size and Location of Shunt Capacitors for Reduction of Losses on Distribution Feeders", IEEE Transaction Power Apparatus and Systems, Vol. 100, pp.1105-1118, March 1981.

16. Grainger J.J. and Lee S.H., "Capacity Release by Shunt Capacitor Placement on Distribution Feeders: A New Voltage-Dependent Model", IEEE Transaction Power Apparatus and Systems, Vol.101, pp.1012-1020, May 1982.

17. Grainger J.J., Lee S.H. and El-Kib A.A., "Design of a Real-Time Switching Control Scheme for Capacitive Compensation of Distribution Feeders", IEEE Transaction Power Apparatus and Systems, Vol.101, pp2420-2428, August 1982.

18. Grainger J.J., Civalar S., Lee S.H., "Optimal Design and Control Scheme for Continuous Capacitive Compensation of

Distribution Feeders", IEEE Transaction Power Apparatus and Systems, Vol.102, pp.3271-3278, October 1983.

19. Lee K.Y. and Pond L.G., "An Optimization Technique For Reactive Power Planning of Subtransmission Network Under Normal Operation, IEEE Transaction on Power Systems, Vol.PWRS-1, No. 2, May 1986.

20. Venkataramana A., Carr J., and Ranshaw R.S., "Optimal Reactive Power Allocation", IEEE Transactions on Power Systems, Vol. PWRS-2, No.1, Febuary 1987.

21. Gonen T., "Electrical Power Distribution System Engineering", McGraw-Hill Series in Electrical Engineering, 1986.

22. Granville S., Periera M.V.P., and Monticelli A., "An Integrated Methodology for VAR Sources Planning, IEEE Transactions on Power Systems, Vol.3, No.2, May 1988.

23. Saied M.M., "Optimal Power Factor Correction", IEEE Transactions on Power Systems, Vol.3, No.3, August 1988.

24. Aoki K., Fan M., and Mishidorl A., "Optimal VAR Planning by Approximation Method for Recursive Mixed-Integer Linear Programming, IEEE Transaction on Power System, Vol.3, No.4, December 1988.

25. ประสิทธิ์ พิทยพัฒน์, "เทคนิคการปรับปรุงตัวประกอบกำลังสำหรับวิศวกรและช่างเทคนิค", หนังสือโครงการไฟฟ้าชุดที่ 1 ของวารสารเทคนิค เครื่องกล-ไฟฟ้า-อุตสาหการ, กันยายน 2529.

ภาคผนวก

```
Program   Optimum_Size_and_Location_Of_Capacitors ;
Uses printer,crt,turbo3;


        {---------------------------------------}
        {                                       }
        {            Program  Opt.Pas I         }
        {      Find Optimum Size and Location   }
        {                  Of                   }
        {     Capacitor in Distribution System  }
        {                                       }
        {---------------------------------------}



label     X1,X2,X3,X4,X5,X6,X7,X8,X9,X10,X11;

type      matrix=array[1..30,1..30] of real;
          matric=array[1..30] of real;
          matint=array[1..30] of integer;
          strg=string[100];

var       n,m,i,j:integer;     {number of buses}
          vsub:real;           {substation voltage}
          dfrom:integer;       {data from}
          resist:matric;       {resistance in ohms per miles}
          react:matric;        {reactance in ohms per miles}
          phyl:matric;         {physical section lenght in miles}
          eqil:matric;         {length of equivalent uniform feeder}
          normal:matric;       {normalized length}
          kw:matric;           {kw load at bus i}
          kvar:matric;         {kvar load at bus i}
          maxkvar:matric;      {maximum kvar load}
          bkvarmax:integer;    {maximum kvar bus}
          sum:matric;          {summation of 1:physical length        }
                               {             2:uniform feeder in miles}
                               {             3:normalize feeder       }
                               {             4:kvar load              }
                               {             5:kw load                }
          snormal:matric;      {sum of normalized uniform feeder}
          yes:string[1];       {answer yes or no}
          filevar:file of real;{data files}
          name:strg;           {file name}
          s:strg;              {print string}
          value:matric;
          rohm:matric;         {resistance in ohms}
          xohm:matric;         {reactance in ohms}
          h:matric;            {normalized bus distance from substation}
          rdist:matric;        {real distance from substation}
          basez:real;          {base impedance}
          basei:real;          {base current}
          aresist:real;        {all resistance in feeder}
          r:real;              {resistance in ohms/mile     }
                               {of equivalent uniform feeder}
          vsubpu:real;         {per unit voltage at substation}
          angsub:real;         {angle of substation}
          vbuspuold:matric;    {per unit bus voltage old}
```

```
angbusold:matric;    {angle of bus old}
vrbuspuold:matric;   {real part of per unit bus voltage old}
vxbuspuold:matric;   {imaginary part of per unit bus voltage old}
vbuspunew:matric;    {per unit bus voltage new}
angbusnew:matric;    {angle of bus new}
vrbuspunew:matric;   {real part of per unit bus voltage new}
vxbuspunew:matric;   {imaginary part of per unit bus voltage new}
pup:matric;          {per unit real power}
puq:matric;          {per unit reactive power}
pur:matric;          {per unit resistance}
pux:matric;          {per unit reactance}
puir:matric;         {per unit real load current}
puix:matric;         {per unit reactive load current}
puilr:matric;        {real part of per unit line current}
puilx:matric;        {imaginary part of per unit line current}
delta:matric;        {different value}
vdiffmax:real;       {maximum different voltage}
fqx:matric;          {Fq(X)}
nc:integer;          {number of installed capacitors}
kvarc:matric;        {kvar of each capacitor}
kvarcpu:matric;      {per unit kvar of each capacitor}
puirc:matric;        {real part of per unit      }
                     {load current with capacitor}
puixc:matric;        {imaginary part of per unit }
                     {load current with capacitor}
busc:matint;         {bus that capacitor is installed}
hc:matric;           {normalized distance of capacitor}
                     {from substation                  }
kp:real;             {annual cost per unit of power loss}
                     {in $/kw/yr                        }
kc:real;             {anual cost per unit of installed}
                     {capacitor in $/kvar three phases}
hcmatric:matrix;     {2d matric of normalized distance}
icq:matric;          {current of each capacitor}
int:matric;          {integrate fq(x)dx}
k:real;              {constance}
a:matrix;            {hcmatric+intmatric}
opticq:matric;       {optimum capacitor current}
ficq:matric;         {1/2Icq + sigma Icq rest}
optbus:matint;       {optimum bus distance of  }
                     {capacitor from substation}
icrpu:matric;        {per unit real current of each capacitor}
icxpu:matric;        {per unit imaginary current of each capacitor}
diffic:matric;       {difference between optimum capacitor      }
                     {current and the previous capacitor current}
difficmax:real;      {maximum difference of capacitor current}
buscold:matric;      {capacitor bus old}
sopticq:matric;      {sum of optimum Icq}
sopticd:matric;      {sum of optimum Icd}
plossq:real;         {power loss in q-axis}
plossd:real;         {power loss in d-axis}
plossbc:real;        {power loss before install capacitors}
p:real;              {power}
dvr:matric;          {different of real voltage p-q}
dvx:matric;          {different of imaginary voltage p-q}
dv:matric;           {different voltage}
dang:matric;         {different angle}
rpu:matric;          {resistance in per unit}
xpu:matric;          {reactance in per unit}
zpu:matric;          {per unit impedance}
```

```
          zang:matric;        {angle of impedance}
          spq:matric;         {power flow p-q}
          sangpq:matric;      {angle of power flow p-q}
          sqp:matric;         {power flow q-p}
          sangqp:matric;      {angle of power flow q-p}
          spqr:matric;        {power flow p-q real}
          spqx:matric;        {power flow p-q imaginary}
          sqpr:matric;        {power flow q-p real}
          sqpx:matric;        {power flow q-p imaginary}
          plossp:real;        {real power loss}
          plosq:real;         {reactive power loss}
          plosspb:real;       {real power loss before install capacitor}
          plosqb:real;        {reactive power loss befor install capacitor}
          dplossp:real;       {real power loss reduction}
          dplossq:real;       {reactive power loss reduction}
          plq:real;           {power due to q-axis}
          costc:real;         {obj. fn. of capacitor cost}
          costpl:real;        {obj. fn. of power loss cost}
          totalcost:real;     {obj. fn. = costc+costpl}
          cost1,cost:real;

{****************}
{* MAKING SOUND *}
{****************}
procedure beep;

begin
  sound(100);delay(50);
  sound(200);delay(50);
  sound(300);delay(50);
  sound(400);delay(50);
  sound(500);delay(50);
  nosound;
end;


{**************************}
{* PRINT TITLE OF PROGRAM *}
{**************************}
Procedure ptitle;

begin
  clrscr;
  gotoxy(18,3); write('****************************************');
  gotoxy(18,4); write('*                                      *');
  gotoxy(18,5); write('*              Optc I                  *');
  gotoxy(18,6); write('*     Optimum Size And Location        *');
  gotoxy(18,7); write('*                Of                    *');
  gotoxy(18,8);write('*  Capacitors In Distribution System   *');
  gotoxy(18,9);write('*            Writen By                 *');
  gotoxy(18,10);write('*     Ass.Prof. Prasit Pittiyapat      *');
  gotoxy(18,11);write('*                And                   *');
  gotoxy(18,12);write('*         Mr. Parames Chutima          *');
  gotoxy(18,13);write('*      Chulalongkorn University        *');
  gotoxy(18,14);write('*                                      *');
  gotoxy(18,15);write('****************************************');
  gotoxy(18,22);write('   <<<  Press Any Key To Continue  >>> ');
  readln;
  beep;
end;
```

```
{*********************************}
{* FIND SUMMATION OF INPUT DATA *}
{*********************************}
Procedure sum12345(var n:integer;
                   var sum,phyl,eqil,resist,normal,kvar,kw:matric;
                   var bkvarmax:integer);
var i:integer;

begin
  { SUM OF PHYSICAL LENGTH }
  sum[1]:=0.0;
  for i:=1 to n do
    sum[1]:=sum[1]+phyl[i];

  { SUM OF EQUIVALENT UNIFORM FEEDER IN MILES }
  for i:=1 to n do
    eqil[i]:=resist[i]*phyl[i]/resist[bkvarmax];
  sum[2]:=0.0;
  for i:=1 to n do
    sum[2]:=sum[2]+eqil[i];

  { SUM OF NORMALIZED UNIFORM FEEDER }
  for i:=1 to n do
    normal[i]:=eqil[i]/sum[2];
  sum[3]:=0.0;
  for i:=1 to n do
    sum[3]:=sum[3]+normal[i];

  { SUM OF KVAR }
  sum[4]:=0.0;
  for i:=1 to n do
    sum[4]:=sum[4]+kvar[i];

  { SUM OF KW }
  sum[5]:=0.0;
  for i:=1 to n do
    sum[5]:=sum[5]+kw[i];
  clrscr;
end;

{*********************************}
{* PRINT SUMMATION OF INPUT DATA *}
{*********************************}
Procedure psum12345(var n:integer;
                    var resist,phyl,eqil,normal,kw,kvar,sum:matric);
var i:integer;

begin
  clrscr;
  writeln('                      **********************');
  writeln('                      * TABLE OF INPUT DATA *');
  writeln('                      **********************');
  writeln;
  writeln(' |-----|----------|----------|--------------|--------------|--------|--------|');
  writeln(' |     |RESISTANCE| PHYSICAL |SECTION LENGTH|NORMALIZED    |        |        |');
  writeln(' | BUS |OHMS/MILE | SECTION  |OF EQUIVALENT |SECTION LENGTH|   KW   |  KVAR  |');
  writeln(' | No. |          | LENGTH   |UNIFORM FEEDER|FOR EQUIVALENT|  LOAD  |  LOAD  |');
  writeln(' |     |          | IN MILES |IN MILES      |UNIFORM FEEDER|        |        |');
```

```
  writeln(' ¦-----¦----------¦----------¦--------------¦--------------¦--------¦--------¦');

  for i:=1 to n do
  writeln(' ¦',i:3,'  ¦',resist[i]:9:5, ' ¦',phyl[i]:9:5,
          ' ¦',eqil[i]:11:5,'   ¦',normal[i]:11:5,
          '   ¦',kw[i]:8:1,'¦',kvar[i]:8:1,'¦');
  writeln(' ¦-----¦----------¦----------¦--------------¦--------------¦--------¦--------¦');
  writeln(' ¦     total      ¦',sum[1]:9:5,' ¦',sum[2]:11:5,
          '   ¦',sum[3]:11:5,'   ¦',sum[5]:8:1,'¦',sum[4]:8:1,'¦');
  writeln(' ¦---------------¦----------¦--------------¦--------------¦--------¦--------¦');
  writeln;
  writeln('                      <<<  Press Any Key To Continue  >>>                  ');
  readln;
  clrscr;
end;

{*****************************}
{* CHANGE FROM REAL TO POLAR *}
{*****************************}
Procedure realpolar(n:integer;r,x:matric;var z,ang:matric);
var i:integer;

begin
for i:= 1 to n do
  begin
    z[i]:=sqrt(sqr(r[i])+sqr(x[i]));
    ang[i]:=arctan(x[i]/r[i]);
  end;
end;

{*****************************}
{* CHANGE FROM POLAR TO REAL *}
{*****************************}
Procedure polarreal(n:integer;z,ang:matric;var r,x:matric);
var i:integer;

begin
for i:=1 to n do
  begin
    r[i]:=z[i]*cos(ang[i]);
    x[i]:=z[i]*sin(ang[i]);
  end;
end;

{*******************}
{* PRINT TABLE R&X *}
{*******************}
Procedure prx(n:integer;
              rohm,xohm:matric);
var i:integer;

begin
  clrscr;
  writeln('                    **********************************');
  writeln('                    * VALUE OF R & X FROM BUS TO BUS *');
  writeln('                    **********************************');
  writeln;
  writeln('                   ¦---------¦-----------¦-----------¦');
  writeln('                   ¦   BUS   ¦ RESISTANCE ¦ REACTANCE ¦');
  writeln('                   ¦         ¦   IN OHMS  ¦  IN OHMS   ¦');
```

```
  writeln('                    ¦---------¦------------¦------------¦');
  for i:= 1 to n do
  writeln('                    ¦',i:3,' -',i+1:2,'  ¦',rohm[i]:10:5,'  ¦',
          xohm[i]:10:5,'  ¦');
  writeln('                    ¦---------¦------------¦------------¦');
  writeln;
  writeln('                     <<<  Press Any Key To Continue  >>> ');
  readln;
end;

{**************************}
{* PRINT TABLE OF DISTANCE *}
{**************************}
Procedure pdistance(n:integer;
                    h,rdis:matric);
var i:integer;

begin
  clrscr;
  writeln('                     *******************************');
  writeln('                     * NORMALIZED & REAL  DISTANCE *');
  writeln('                     *******************************');
  writeln;
  writeln('                    ¦-----¦------------¦------------¦');
  writeln('                    ¦ BUS ¦ NORMALIZED ¦    REAL    ¦');
  writeln('                    ¦     ¦  DISTANCE  ¦  DISTANCE  ¦');
  writeln('                    ¦-----¦------------¦------------¦');
  for i:= 1 to n do
  writeln('                    ¦',i:3,'  ¦',h[i]:10:6,'  ¦',rdist[i]:10:4,'  ¦');
  writeln('                    ¦-----¦------------¦------------¦');
  writeln;
  writeln('                   <<<  Press Any Key To Continue  >>>');
  readln;
end;

{************************************************************}
{ SUBPROGRAM FIND RADIAL LOAD FLOW BEFORE INSTALL CAPACITORS }
{************************************************************}
Procedure loadflow(var n:integer;
                   var vbuspuold,angbusold,pup,puq,puir,puix,
                       puilr,puilx,pur,pux,vrbuspuold,vxbuspuold,
                       vrbuspunew,vxbuspunew,vbuspunew,
                       angbusnew:matric;
                   var vsubpu:real);
label     P1;
var       i,j:integer;
          delta:matric;
          vdiffmax:real;

begin
clrscr;
P1: { FIND PER UNIT LOAD CURRENT AT EACH BUS }
  for i:=1 to n do
  begin
    puir[i]:=(pup[i]*cos(angbusold[i])+puq[i]*sin(angbusold[i]))/vbuspuold[i];
    puix[i]:=(pup[i]*sin(angbusold[i])-puq[i]*cos(angbusold[i]))/vbuspuold[i];
  end;

  { FIND LINE CURRENT }
  for i:=1 to n do
```

```
 begin
   puilr[i]:=0;
   puilx[i]:=0;
 end;
 for i:=1 to n do
 for j:=1 to i do
 begin
   puilr[i]:=puilr[i]+puir[j];
   puilx[i]:=puilx[i]+puix[j];
 end;

 { FIND NEW BUS VOLTAGE  Vr+jVx }
 vrbuspunew[n]:=vsubpu-(puilr[n]*pur[n]-puilx[n]*pux[n]);
 vxbuspunew[n]:=-(puilr[n]*pux[n]+puilx[n]*pur[n]);
 for i:=n-1 downto 1 do
 begin
   vrbuspunew[i]:=vrbuspuold[i+1]-(puilr[i]*pur[i]-puilx[i]*pux[i]);
   vxbuspunew[i]:=vxbuspuold[i+1]-(puilr[i]*pux[i]+puilx[i]*pur[i]);
 end;
 clrscr;

 { FIND NEW BUS VOLTAGE AND ANGLE }
 for i:=1 to n do
 begin
   vbuspunew[i]:=sqrt(sqr(vrbuspunew[i])+sqr(vxbuspunew[i]));
   angbusnew[i]:=arctan(vxbuspunew[i]/vrbuspunew[i]);
 end;

 { FIND DIFFERENCE OF NEW AND OLD ITERATION }
 for i:=1 to n do
   delta[i]:=abs(vbuspuold[i]-vbuspunew[i]);

 { FIND MAXIMUM DIFFERENCE }
 vdiffmax:=delta[1];
 for i:=2 to n do
 begin
   if vdiffmax < delta[i] then
     vdiffmax:=delta[i];
 end;
 while vdiffmax > 0.0001 do
 begin
   for i:=1 to n do
     begin
       vbuspuold[i]:=vbuspunew[i];
       angbusold[i]:=angbusnew[i];
       vrbuspuold[i]:=vrbuspunew[i];
       vxbuspuold[i]:=vxbuspunew[i];
     end;
 gotoxy (27,10);
 writeln('<< Load Flow Iteration >> ');
 goto P1;
 end;
end;

{*************************************************}
{ SUBPROGRAM LOAD FLOW AFTER CAPACITORS IS INSTALLED }
{*************************************************}
Procedure cloadflow(var n,nc:integer;
                    var vbuspuold,angbusold,pup,puq,puir,puix,
                        puilr,puilx,pur,pux,vrbuspuold,vxbuspuold,
```

```
                    vrbuspunew,vxbuspunew,vbuspunew,angbusnew,
                    kvarcpu,puirc,puixc,icrpu,icxpu:matric;
                var busc:matint;
                var vsubpu:real);
label     P1;
var       i,j:integer;
          delta:matric;
          vdiffmax:real;

begin
{ SET SUBSTATION BUS VOLTAGE AND ANGLE }
  vsubpu:=1;
  angsub:=0;

{ SET INITIAL BUS VOLTAGE AND ANGLE }
  for i:=1 to n do
  begin
    vbuspuold[i]:=vsubpu;
    angbusold[i]:=angsub;
    vrbuspuold[i]:=1;
    vxbuspuold[i]:=0;
  end;
clrscr;

P1: { FIND PER UNIT LOAD CURRENT AT EACH BUS }
  for i:=1 to n do
  begin
    puir[i]:=(pup[i]*cos(angbusold[i])+puq[i]*sin(angbusold[i]))/vbuspuold[i];
    puix[i]:=(pup[i]*sin(angbusold[i])-puq[i]*cos(angbusold[i]))/vbuspuold[i];
  end;

  { CORRECT PER UNIT LOAD CURRENT AT BUS THAT HAVE CAPACITOR }
  for i:=1 to n do
  begin
    puirc[i]:=puir[i];
    puixc[i]:=puix[i];
  end;
  for i:=1 to nc do
  begin
    icrpu[i]:= -kvarcpu[i]*vbuspuold[busc[i]]*sin(angbusold[busc[i]]);
    icxpu[i]:= +kvarcpu[i]*vbuspuold[busc[i]]*cos(angbusold[busc[i]]);
    puirc[busc[i]]:=puirc[busc[i]]-kvarcpu[i]*vbuspuold[busc[i]]*sin(angbusold[busc[i]]);
    puixc[busc[i]]:=puixc[busc[i]]+kvarcpu[i]*vbuspuold[busc[i]]*cos(angbusold[busc[i]]);
  end;

  { FIND LINE CURRENT }
  for i:=1 to n do
  begin
    puilr[i]:=0;
    puilx[i]:=0;
  end;
  for i:=1 to n do
  for j:=1 to i do
  begin
    puilr[i]:=puilr[i]+puirc[j];
    puilx[i]:=puilx[i]+puixc[j];
  end;

  { FIND NEW BUS VOLTAGE  Vr+jVx }
  vrbuspunew[n]:=vsubpu-(puilr[n]*pur[n]-puilx[n]*pux[n]);
```

```
 vxbuspunew[n]:=-(puilr[n]*pux[n]+puilx[n]*pur[n]);
 for i:=n-1 downto 1 do
 begin
   vrbuspunew[i]:=vrbuspuold[i+1]-(puilr[i]*pur[i]-puilx[i]*pux[i]);
   vxbuspunew[i]:=vxbuspuold[i+1]-(puilr[i]*pux[i]+puilx[i]*pur[i]);
 end;
 clrscr;

 { FIND NEW BUS VOLTAGE AND ANGLE }
 for i:=1 to n do
 begin
   vbuspunew[i]:=sqrt(sqr(vrbuspunew[i])+sqr(vxbuspunew[i]));
   angbusnew[i]:=arctan(vxbuspunew[i]/vrbuspunew[i]);
 end;

 { FIND DIFFERENCE OF NEW AND OLD ITERATION }
 for i:=1 to n do
   delta[i]:=abs(vbuspuold[i]-vbuspunew[i]);

 { FIND MAXIMUM DIFFERENCE }
 vdiffmax:=delta[1];
 for i:=2 to n do
 begin
   if vdiffmax < delta[i] then
     vdiffmax:=delta[i];
 end;
 while vdiffmax > 0.0001 do
 begin
   for i:=1 to n do
     begin
       vbuspuold[i]:=vbuspunew[i];
       angbusold[i]:=angbusnew[i];
       vrbuspuold[i]:=vrbuspunew[i];
       vxbuspuold[i]:=vxbuspunew[i];
     end;
 gotoxy (27,10);
 writeln('<< Load Flow Iteration >>');
 goto P1;
 end;
end;

{*****************************}
{* PRINT RESULT OF LOAD FLOW *}
{*****************************}
Procedure ploadflow(var n:integer;
                    var vbuspunew,angbusnew:matric);
var i:integer;

begin
  clrscr;
  writeln('                        **********************       ');
  writeln('                        * RESULT OF LOAD FLOW *       ');
  writeln('                        **********************       ');
  writeln;
  writeln('                    ¦-----¦-------------¦-------------¦');
  writeln('                    ¦ BUS ¦   PU VOLT   ¦    PU ANG   ¦');
  writeln('                    ¦-----¦-------------¦-------------¦');
  for i:=1 to n do
  writeln('                    ¦',i:3,'  ¦',vbuspunew[i]:12:8,' ¦',angbusnew[i]:12:8,' ¦');
  writeln('                    ¦-----¦-------------¦-------------¦');
```

```
end;

{*******************}
{* PRINT LINE FLOW *}
{*******************}
Procedure plineflow(var n:integer;
                    var puilr,puilx:matric);
var i:integer;

begin
  clrscr;
  writeln('                              ************              ');
  writeln('                              * LINE FLOW *              ');
  writeln('                              ************              ');
  writeln;
  writeln('                     ¦----------¦-------------¦-------------¦');
  writeln('                     ¦    BUS   ¦    PU Ir    ¦    PU Ix    ¦');
  writeln('                     ¦----------¦-------------¦-------------¦');
  for i:=1 to n do
  writeln('                     ¦',i:3,' -',i+1:3,'  ¦',puilr[i]:12:8,
                              ' ¦',puilx[i]:12:8,' ¦');
  writeln('                     ¦----------¦-------------¦-------------¦');
end;

{************************}
{ SUBPROGRAM FIND Fq(X) }
{************************}
Procedure findfqx(var n:integer;
                  var angbusnew,puix,fqx:matric);
var i,j:integer;
    sumangle:real;
    product:matric;

begin
  sumangle:=0;
  for i:=1 to n do
    sumangle:=sumangle+sin(angbusnew[i]);
  sumangle:=sumangle/n;
  for i:=1 to n do
    product[i]:=-(sumangle*puix[i]/sin(angbusnew[i]));
  for i:=1 to n do
    fqx[i]:=0;
  for i:=1 to n do
  for j:=1 to i do
    fqx[i]:=fqx[i]+product[j];
end;

{***************}
{* PRINT Fq(X) *}
{***************}
Procedure pfqx(var n:integer;
               normal,snormal,fqx:matric);
var i:integer;

begin
  clrscr;
  writeln('                                 ******************            ');
  writeln('                                 * TABLE OF Fq(X) *            ');
  writeln('                                 ******************            ');
  writeln;
```

```
  writeln('                    ¦------------------------¦--------------¦');
  writeln('                    ¦   NORMALIZED DISTANCE   ¦     Fq(X)     ¦');
  writeln('                    ¦------------------------¦--------------¦');
  for i:=1 to n-1 do
  writeln('                    ¦',snormal[i]:11:8,' - ',snormal[i+1]:10:8,
                      ' ¦',fqx[i]:14:10,' ¦');
  writeln('                    ¦',snormal[n]:11:8,' - 0.00000000 ¦',
                         fqx[n]:14:10,' ¦');
  writeln('                    ¦------------------------¦--------------¦');
  writeln;
  writeln('                        <<<  Press Any Key To Continue  >>>    ');
  readln;
end;

{*********************}
{ SUBPROGRAM FIND Icq }
{*********************}
Procedure findicq(var nc:integer;
                  var a:matrix);
var k,l,m,t,p:integer;
    ri:real;

begin
  m:=nc+1;
  for i:=1 to nc do
  begin
    if a[i,i]=0 then
    begin
      for k:=i+1 to nc do
      begin
        if a[k,i] <> 0 then
        begin
          for l:=1 to m do
          begin
            ri:=a[i,l];
            a[i,l]:=a[k,l];
            a[k,l]:=ri;
          end;
        end;
      end;
    end;
  if a[i,i] <> 1 then
  begin
    ri:=1/a[i,i];
    for t:=i to m do
      a[i,t]:=a[i,t]*ri;
  end;
  for p:=1 to nc do
  begin
    if (p<>i) and (a[p,i]<>0.0) then
    begin
      ri:=a[p,i];
      for t:=i to m do
        a[p,t]:=a[p,t]-a[i,t]*ri;
    end;
  end;
  end;
end;

{*************************************}
```

```
{ SUBPROGRAM B: FIND Hi WHEN KNOWN Icq }
{************************************}
Procedure findh(var n,nc:integer;
                var ficq,fqx,snormal:matric;
                var optbus:matint);
var i,j:integer;

begin
  for i:=1 to nc do
  for j:=1 to n-1 do
  begin
    if ficq[i]<fqx[1] then
      optbus[i]:=1
    else
      if (ficq[i]>fqx[j]) and (ficq[i]<fqx[j+1]) then
        optbus[i]:=j+1;
  end;
end;

{*********************************}
{ WRITE OPTIMUM SIZE AND LOCATION }
{*********************************}
Procedure presult(var optbus:matint;
                  var nc:integer;
                  var kvarcpu:matric;
                  var sum:matric);

var k:matric;
begin;
  for i:=1 to nc do
    k[i]:=kvarcpu[i]*sum[4];
  clrscr;
  writeln('                          ******************************');
  writeln('                          * OPTIMUM SIZE AND LOCATION  *');
  writeln('                          ******************************');
  writeln;
  writeln('                       ¦-----¦-----------------------------¦');
  writeln('                       ¦ BUS ¦   OPTIMUM KVAR OF CAPACITOR  ¦');
  writeln('                       ¦-----¦-----------------------------¦');
  for i:=1 to nc do
  begin
    writeln('                        ¦',optbus[i]:3,'  ¦',k[i]:20:10,
                                      '          ¦');
    writeln('                        ¦-----¦-----------------------------¦');
  end;
  writeln;
  writeln('                       <<<  Press Any Key To Continue  >>>  ');
  readln;
end;

{*******************}
{* PRINT P&Q LOSSES *}
{*******************}
Procedure ploss(n:integer;
                plosspb,plossp,dplossp,plosqb,plosq,dplossq,costc,
                costpl,totalcost:real);

begin
clrscr;
writeln('                        *************************     ');
```

```
writeln('                          *  TABLE OF POWER LOSSES *      ');
writeln('                          ************************       ');
writeln;
writeln('                     ¦----------¦----------¦------------¦');
writeln('                     ¦  Before  ¦   After  ¦   Losses   ¦');
writeln('                     ¦ Installed ¦ Installed ¦  Reduction ¦');
writeln('                     ¦ Capacitor ¦ Capacitor ¦  (kw/kvar) ¦');
writeln('              ¦--------¦----------¦----------¦------------¦');
writeln('              ¦ P Loss ¦',plosspb:9:2,' ¦',plossp:9:2,' ¦',
                       dplossp:11:2,' ¦');
writeln('              ¦--------¦----------¦----------¦------------¦');
writeln('              ¦ Q Loss ¦',plosqb:9:2,' ¦',plosq:9:2,' ¦',
                       dplossq:11:2,' ¦');
writeln('              ¦--------¦----------¦----------¦------------¦');
writeln('              ¦       Capacitor Cost ($)      ¦',costc:11:1,' ¦');
writeln('              ¦-------------------------------¦------------¦');
writeln('              ¦       Power Loss Cost ($)     ¦',costpl:11:1,' ¦');
writeln('              ¦-------------------------------¦------------¦');
writeln('              ¦       Objective Function ($)  ¦',totalcost:11:1,' ¦');
writeln('              ¦-------------------------------¦------------¦');
writeln;
writeln('                     <<<  Press Any Key To Continue  >>>     ');
readln;
end;

{*****************************}
{* FIND POWER DUE TO Q-AXIS *}
{*****************************}
Procedure pqaxis(var nc,n:integer;
                 var sopticq,opticq,snormal,fqx:matric;
                 var optbus:matint;
                 var plq:real);
var i,j:integer;
begin
for i:=1 to nc do
  sopticq[i]:=0;
for i:=1 to nc do
for j:=1 to i  do
 sopticq[i]:=sopticq[i]+opticq[j];
plq:=0;
snormal[n+1]:=0;
for i:=n downto optbus[nc] do
begin
  plq:=plq+(snormal[i]-snormal[i+1])*sqr(fqx[i]-sopticq[nc]);
end;
m:=nc;
for i:=nc-1 downto 1 do
begin
  for j:=optbus[m]-1 downto optbus[i] do
    plq:=plq+(snormal[j]-snormal[j+1])*sqr(fqx[j]-sopticq[i]);
  m:=m-1;
end;
if optbus[1] <> 1 then
begin
  i:=optbus[1]-1;
  for j:=i downto 1 do
    plq:=plq+sqr(fqx[j])*(snormal[j]-snormal[j+1]);
end;
plq:=3*r*plq*sum[4];
end;
```

```
{*************************************************************************}

  {*********************************}
  {      MAIN PROGRAM OPT.PAS       }
  {*********************************}
  begin
    clrscr;
    beep;

    { PRINT TITLE }
    ptitle;

    { GETTING INPUT DATA }
    clrscr;
    gotoxy(20,8);writeln('NUMBER OF BUSES = ');
    gotoxy(38,8);readln(n);
    gotoxy(20,9);writeln('VOLTAGE AT SUBSTATION IS = ');
    gotoxy(52,9);writeln('KV');
    gotoxy(48,9);readln(vsub);
    gotoxy(20,10);writeln('DO YOU WANT TO GET DATA FROM ');
    gotoxy(20,11);writeln('1) DATA FILES');
    gotoxy(20,12);writeln('2) INPUT DATA FROM KEYBOARD');
    gotoxy(20,13);writeln('(TYPE 1 OR 2) =======> ');
    x6:gotoxy(43,13);readln(dfrom);
    if dfrom<>1 then
    if dfrom<>2 then
    begin
      beep;
      gotoxy(20,14);
      writeln(' ERROR  YOU MUST TYPE 1 OR 2 PRESS ANY KEY TO TRY AGAIN ');
      readln;
      gotoxy(20,14);
      writeln('                                                          ');
      gotoxy(20,13);writeln('(TYPE 1 OR 2) =======>  ');
      goto X6;
    end;
    if dfrom=2 then goto X1;
      X7:clrscr;

    { GETTING INPUT DATA FROM DATA FILES }
    { RESIST.DAT & REAC.DAT & PHYL.DAT & KW.DAT & KVAR.DAT }
      write('DO YOU WANT TO CHANGE DATA IN DATA FILES (Y/N)? ');
      readln(yes);
      if yes <> 'Y' then
      if yes <> 'N' then
      if yes <> 'y' then
      if yes <> 'n' then
      begin
        beep;
        writeln(' ERROR YOU MUST TYPE Y/N PRESS ANY KEY TO TRY AGAIN ');
        readln;
        goto X7;
      end;
      if (yes = 'Y') or (yes = 'y') then

    { CHANGE DATA IN DATA FILES }
      begin
```

```
    writeln('DATA FILES THAT YOU CAN CHANGE IS ');
    writeln('1) RESIST.DAT');
    writeln('2) REACT.DAT');
    writeln('3) PHYL.DAT');
    writeln('4) KW.DAT');
    writeln('5) KVAR.DAT');
    write('TYPE A FILE NAME THAT YOU WANT TO CHANGE ===> ');
    X2: readln(name);
    assign(filevar,name);
    rewrite(filevar);
    for i:= 1 to n do
    begin
      write('INPUT ',name,' NUMBER (',i,') = ');
      readln(value[i]);
      write(filevar,value[i]);
    end;
    close(filevar);
    write('DO YOU WANT TO CHANGE ANOTHER FILES (Y/N)? ');
    readln(yes);
    if (yes = 'Y') or (yes = 'y') then
    begin
      writeln('DATA FILES THAT YOU CAN CHANGE IS ');
      writeln('1) RESIST.DAT');
      writeln('2) REACT.DAT');
      writeln('3) PHYL.DAT');
      writeln('4) KW.DAT');
      writeln('5) KVAR.DAT');
      write('TYPE A FILE NAME THAT YOU WANT TO CHANGE ===> ');
      goto X2;
    end;
  end;

{ READ DATA FROM DATA FILES }
  assign(filevar,'resist.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,resist[i]);
  close(filevar);
  assign(filevar,'react.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,react[i]);
  close(filevar);
  assign(filevar,'phyl.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,phyl[i]);
  close(filevar);
  assign(filevar,'kw.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,kw[i]);
  close(filevar);
  assign(filevar,'kvar.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,kvar[i]);
  close(filevar);
  goto X3;
```

```
{ INPUT DATA FROM KEYBOARD }
  X1: clrscr;
  writeln('THE FIRST BUS IS AT THE END OF FEEDER');
  for i:=1 to n do
  begin
    writeln('****  BUS NUMBER (',i,')',' ****');
    write('RESISTANCE IN OHM/MILES = ');readln(resist[i]);
    write('REACTANCE IN OHM/MILES = ');readln(react[i]);
    write('LENGTH IN MILES = ');readln(phyl[i]);
    write('THREE PHASE KILOWATT LOAD = ');readln(kw[i]);
    write('THREE PHASE KILOVAR LOAD = ');readln(kvar[i]);
    writeln;
  end;

{ FIND MAXIMUM KVAR }
X3: maxkvar[1]:=kvar[1];
bkvarmax:=1;                    {find maximum kvar bus}
for i:=2 to n do
  if maxkvar[1]<kvar[i] then
  begin
    maxkvar[1]:=kvar[i];
    bkvarmax:=i;
  end;

{ FIND SUMMATION OF DATA OF FEEDER }
sum12345(n,sum,phyl,eqil,resist,normal,kvar,kw,bkvarmax);

{ WRITE RESULT FROM CALCULATION }
psum12345(n,resist,phyl,eqil,normal,kw,kvar,sum);
beep;

{ FIND BUS TO BUS RESISTANCES }
for i:= 1 to n do
rohm[i]:=resist[i]*phyl[i];

{ FIND BUS TO BUS REACTANCES }
for i:= 1 to n do
xohm[i]:=react[i]*phyl[i];

{ WRITE R & X BETWEEN BUS }
prx(n,rohm,xohm);
beep;

{ NORMALIZED EQUIVALENT DISTANCE H(X) }
for i:= 1 to n do
  h[i]:=0.0;
for i:= 1 to n do
for j:= i to n do
  h[i]:=h[i]+normal[j];

{ REAL DISTANCE OF BUS FROM SUBSTATION }
for i:= 1 to n do
  rdist[i]:=0.0;
for i:= 1 to n do
for j:= i to n do
  rdist[i]:=rdist[i]+phyl[j];

{ SUM OF NORMALIZED DISTANCES }
for i:=1 to n do
  snormal[i]:=0;
```

```
for i:=1 to n do
for j:=n downto i do
  snormal[i]:=snormal[i]+normal[j];

{ WRITE NORMALIZED & REAL DISTANCE }
pdistance(n,h,rdist);
beep;

{ FIND BASE IN THIS PROGRAM }
basez:=sqr(vsub)*1000/sum[4];
basei:=sum[4]/(sqrt(3)*vsub);
aresist:=0.0;
for i:= 1 to n do
aresist:=aresist+resist[i]*phyl[i];
r:=aresist/sum[2];
clrscr;
writeln('{***************************}');
writeln('{ BASE VALUE IN THIS PROBLEM }');
writeln('{***************************}');
writeln;
writeln('BASE VOLTAGE (KV)      = ',vsub:12:4);
writeln('BASE POWER (KVA)       = ',sum[4]:12:4);
writeln('BASE CURRENT (AMP)     = ',basei:12:4);
writeln('BASE IMPEDANCE (OHMS)  = ',basez:12:4);
writeln;
writeln('RESISTANCE IN OHM/MILES OF EQUIVALENT UNIFORM FEEDER (r) = ',r:10:5);
writeln;
writeln('<<<  Press Any Key To Continue  >>>');
readln;beep;

{ FIND PER UNIT  P,Q,R,X  FOR EACH BUS }
for i:=1 to n do
begin
  pup[i]:=kw[i]/sum[4];
  puq[i]:=kvar[i]/sum[4];
  pur[i]:=resist[i]*phyl[i]/basez;
  pux[i]:=react[i]*phyl[i]/basez;
end;

{ FIND PER UNIT RESISTANCES }
for i:=1 to n do
begin
  rpu[i]:=rohm[i]/basez;
  xpu[i]:=xohm[i]/basez;
end;
realpolar(n,rpu,xpu,zpu,zang);

{*****************************************************}
{ SUBPROGRAM RADIAL LOAD FLOW BEFORE INSTALL CAPACITORS }
{*****************************************************}
{ SET SUBSTATION BUS VOLTAGE AND ANGLE }
vsubpu:=1;
angsub:=0;

{ SET INITIAL BUS VOLTAGE AND ANGLE }
for i:=1 to n do
begin
  vbuspuold[i]:=vsubpu;
  angbusold[i]:=angsub;
  vrbuspuold[i]:=1;
```

```
  vxbuspuold[i]:=0;
end;

{ FIND RADIAL LOAD FLOW }
loadflow(n,vbuspuold,angbusold,pup,puq,puir,puix,
         puilr,puilx,pur,pux,vrbuspuold,vxbuspuold,
         vrbuspunew,vxbuspunew,vbuspunew,angbusnew,
         vsubpu);

{ PRINT LOAD FLOW }
ploadflow(n,vbuspunew,angbusnew);
writeln;
writeln('                This is Load Flow Before Install Capacitors');
writeln;
writeln('                     <<<  Press Any Key To Continue  >>>');
readln;beep;

{ PRINT LINE FLOW }
plineflow(n,puilr,puilx);
writeln;
writeln('                   This is Line Flow Before Install Capacitors');
writeln;
writeln('                        <<<  Press Any Key To Continue  >>>   ');
readln;beep;

{*********************************************}
{* FIND POWER LOSSES BEFORE INSTALL CAPACITORS *}
{*********************************************}
clrscr;
for i:=1 to n do
begin
  dvr[i]:=0;
  dvx[i]:=0;
end;

{ FIND DIFFERENT BUS VOLTAGE }
for i:=1 to n-1 do
begin
  dvr[i]:=vrbuspunew[i]-vrbuspunew[i+1];
  dvx[i]:=vxbuspunew[i]-vxbuspunew[i+1];
end;
dvr[n]:=vrbuspunew[n]-1;
dvx[n]:=vxbuspunew[n];
realpolar(n,dvr,dvx,dv,dang);

{ FIND POWER FLOW FROM P TO Q }
for i:=1 to n do
begin
  spq[i]:=vbuspunew[i]*dv[i]/zpu[i];
  sangpq[i]:=-angbusnew[i]+dang[i]-zang[i];
end;
for i:=1 to n do
begin
  dvr[i]:=-dvr[i];
  dvx[i]:=-dvx[i];
end;
realpolar(n,dvr,dvx,dv,dang);

{ FIND POWER FLOW FROM Q TO P }
sqp[n]:=dv[n]/zpu[n];
```

```
sangqp[n]:=dang[n]-zang[n];
for i:=n-1 downto 1 do
begin
  sqp[i]:=-vbuspunew[i+1]*dv[i]/zpu[i];
  sangqp[i]:=-angbusnew[i+1]+dang[i]-zang[i];
end;

{ FIND REAL & POLAR OF POWER FLOW }
for i:=1 to n do
begin
  spqr[i]:=spq[i]*cos(sangpq[i]);
  spqx[i]:=spq[i]*sin(sangpq[i]);
  sqpr[i]:=sqp[i]*cos(sangqp[i]);
  sqpx[i]:=sqp[i]*sin(sangqp[i]);
end;

{ FIND REAL POWER LOSS }
plossp:=0;
plosq:=0;
for i:=1 to n do
begin
  plossp:=plossp-spqr[i];
  plossp:=plossp+sqpr[i];
end;

{ FIND REACTIVE POWER LOSS }
for i:=1 to n do
begin
  plosq:=plosq-spqx[i];
  plosq:=plosq+sqpx[i];
end;
plosspb:=plossp*sum[4];
plosqb:=plosq*sum[4];


{**********************************}
{ INPUT VALUE IN THE FIRST ITERATION }
{**********************************}
{ INPUT NUMBER AND SIZE OF EACH CAPACITOR }
clrscr;
writeln('*****************************************************');
writeln('* INPUT DATA OF CAPACITORS IN THE FIRST ITERATION *');
writeln('*****************************************************');
writeln;
writeln('EACH CAPACITOR WILL BE INSTALLED AT EACH BUS');
writeln('THE FIRST BUS IS AT THE END OF THE FEEDER');
write('NUMBER OF INSTALLED CAPACITORS = ');
readln(nc);writeln;
for i:=1 to nc do
begin
  write('KVAR RATING OF CAPACITOR (',I,') = ');
  readln(kvarc[i]);
  kvarcpu[i]:=kvarc[i]/sum[4];
  write('THIS CAPACITOR WILL BE INSTALLED AT BUS NUMBER = ');
  readln(busc[i]);
  writeln;
end;
X8:write('DO YOU WANT TO CHANGE DATA OF CAPACITOR (Y/N)? ');
readln(yes);
if yes = 'Y' then
```

```
if yes <> 'N' then
if yes <> 'y' then
if yes <> 'n' then
begin
  beep;
  writeln(' ERROR YOU MUST TYPE Y OR N  PLEASE TRY AGAIN ');
  writeln;
  goto X8;
end;
if (yes = 'Y') or (yes = 'y') then
begin
  write('CAPACITOR THAT YOU WILL CHANGE IS NUMBER = ');
  readln(i);
  write('KVAR RATING OF CAPACITOR (',I,') = ');



  readln(kvarc[i]);
  kvarcpu[i]:=kvarc[i]/sum[4];
  write('THIS CAPACITOR WILL BE INSTALLED AT BUS NUMBER = ');
  readln(busc[i]);
  writeln;
  goto X8;
end;

{ INPUT COST OF CAPACITORS }
writeln('INPUT CONSTANT COST COEFFICIENT OF CAPACITORS');
write('ANNUAL COST PER UNIT OF POWER LOSS IN $/KW/YR (Kp) = ');
readln(kp);
write('ANNUAL COST PER UNIT OF INSTALLED CAPACITOR IN $/KVAR 3 PHASE (Kc) = ');
readln(kc);
beep;

{*****************************************************}
{ SUBPROGRAM RADIAL LOAD FLOW AFTER INSTALL CAPACITORS }
{*****************************************************}
{ FIND RADIAL LOAD FLOW }
X5: cloadflow(n,nc,vbuspuold,angbusold,pup,puq,puir,puix,
              puilr,puilx,pur,pux,vrbuspuold,vxbuspuold,
              vrbuspunew,vxbuspunew,vbuspunew,angbusnew,
              kvarcpu,puirc,puixc,icrpu,icxpu,busc,
              vsubpu);
clrscr;
gotoxy(28,10);
writeln('Pass.......Load Flow');
beep;

{***********************}
{ SUBPROGRAM FIND Fq(X) }
{***********************}
{ FIND Fq(X) }
 findfqx(n,angbusnew,puix,fqx);
 gotoxy(28,11);
 writeln('Pass.......Find Fq(x)');beep;
 for i:=1 to nc do
   buscold[i]:=busc[i];

{***************************************}
{ SUBPROGRAM A: FIND Icq  WHEN KNOWN Hi }
{***************************************}
```

```
{ INPUT H MATRIC }
X4: for i:=1 to nc do
for j:=1 to nc do
begin
  if i>=j then
    hcmatric[i,j]:=snormal[busc[i]]
  else
    hcmatric[i,j]:=snormal[busc[j]];
end;

{ FIND INTEGRATED OF Fq(X)dx }
for i:=1 to nc do
  int[i]:=0;
snormal[n+1]:=0;
for i:=1 to nc do
for j:=busc[i] to n do
  int[i]:=int[i]+fqx[j]*(snormal[j]-snormal[j+1]);

{ FIND INTEGRATED AND COST MATRIC }
k:=kc/(2*r*kp);
for i:=1 to nc do
  int[i]:=int[i]-k/(vbuspunew[busc[i]]*cos(angbusnew[busc[i]]));

{ FORM NC*(NC+1) MATRIC FOR FIND Icq }
for i:=1 to nc do
begin
  for j:=1 to nc do
    a[i,j]:=hcmatric[i,j];
    a[i,nc+1]:=int[i];
end;

{ FIND Icq(i) }
findicq(nc,a);
for i:=1 to nc do
opticq[i]:=a[i,nc+1];
gotoxy(28,12);
writeln('Pass.......Subprogram A');beep;

{************************************}
{ SUBPROGRAM B: FIND Hi WHEN KNOWN Icq }
{************************************}
{ FIND 1/2*Icq + SIGMA Icq REST }
for i:=1 to nc do
  ficq[i]:=opticq[i]/2;
for i:=2 to nc do
for j:=1 to i-1 do
  ficq[i]:=ficq[i]+opticq[j];

{ FIND Hi }
findh(n,nc,ficq,fqx,snormal,optbus);
gotoxy(28,13);
writeln('Pass.......Subprogram B');beep;

{ COMPARE OPTIMUM BUS FOR CONVERGENCE }
for i:=1 to nc do
begin
  if optbus[i]<>busc[i] then
  begin
    for i:=1 to nc do
      busc[i]:=optbus[i];
```

```
      goto X4;
  end;
end;

{*********************************************************}
{ COMPARE RESULT OF OPTIMUM BUS AND CURRENT OF CAPACITORS }
{*********************************************************}
{ COMPARE OPTIMUM BUS OF EACH CAPACITOR }
for i:=1 to nc do
begin
  if optbus[i]<>buscold[i] then
  begin
    for i:=1 to nc do
    begin
      buscold[i]:=optbus[i];
      kvarcpu[i]:=opticq[i]/(vbuspunew[optbus[i]]*cos(angbusnew[optbus[i]]));
    end;
    gotoxy(28,14);
    writeln('Not Pass...Compare Bus');beep;
    goto X5;
  end;
  gotoxy(28,14);
  writeln('Pass.......Compare Bus');beep;
end;

{ COMPARE OPTIMUM CAPACITOR CURRENT }
for i:=1 to nc do
  diffic[i]:=opticq[i]-icxpu[i];
difficmax:=diffic[1];
for i:=2 to nc do
begin
  if difficmax<diffic[i] then
  difficmax:=diffic[i];
end;
if abs(difficmax) > 0.00001 then
  begin
    for i:=1 to nc do
    begin
      buscold[i]:=optbus[i];
      kvarcpu[i]:=opticq[i]/(vbuspunew[optbus[i]]*cos(angbusnew[optbus[i]]));
    end;
    gotoxy(28,15);
    writeln('Not Pass...Compare Current');beep;
    goto x5;
  end;
gotoxy(28,15);
writeln('Pass.......Compare Current');beep;
beep;

{ PRINT LOADFLOW }
ploadflow(n,vbuspunew,angbusnew);
writeln;
writeln('                      LoadFlow After Install Capacitor ');
writeln;
writeln('                       << Press Any Key to Continue >> ');
readln;beep;

{ PRINT LINE FLOW }
plineflow(n,puilr,puilx);
writeln;
```

```
writeln('                          LineFlow After Install Capacitor ');
writeln;
writeln('                           << Press Any Key to Continue >> ');
readln;beep;

{ PRINT Fq(x) }
pfqx(n,normal,snormal,fqx);beep;

{ PRINT RESULT }
presult(optbus,nc,kvarcpu,sum);
beep;

{*****************************************}
{ FIND POWER LOSS AFTER INSTALL CAPACITORS }
{*****************************************}
{ SET DIFFERENT VOLTAGE }
for i:=1 to n do
begin
  dvr[i]:=0;
  dvx[i]:=0;
end;

{ FIND DIFFERENT BUS VOLTAGE }
for i:=1 to n-1 do
begin
  dvr[i]:=vrbuspunew[i]-vrbuspunew[i+1];
  dvx[i]:=vxbuspunew[i]-vxbuspunew[i+1];
end;
dvr[n]:=vrbuspunew[n]-1;
dvx[n]:=vxbuspunew[n];
realpolar(n,dvr,dvx,dv,dang);

{ FIND POWER FLOW FROM P TO Q }
for i:=1 to n do
begin
  spq[i]:=vbuspunew[i]*dv[i]/zpu[i];
  sangpq[i]:=-angbusnew[i]+dang[i]-zang[i];
end;

{ FIND POWER FLOW FROM Q TO P }
for i:=1 to n do
begin
  dvr[i]:=-dvr[i];
  dvx[i]:=-dvx[i];
end;
realpolar(n,dvr,dvx,dv,dang);
sqp[n]:=dv[n]/zpu[n];
sangqp[n]:=dang[n]-zang[n];
for i:=n-1 downto 1 do
begin
  sqp[i]:=vbuspunew[i+1]*dv[i]/zpu[i];
  sangqp[i]:=-angbusnew[i+1]+dang[i]-zang[i];
end;

{ FIND REAL & REACTIVE POWER FLOW }
for i:=1 to n do
begin
  spqr[i]:=spq[i]*cos(sangpq[i]);
  spqx[i]:=spq[i]*sin(sangpq[i]);
  sqpr[i]:=sqp[i]*cos(sangqp[i]);
```

```
  sqpx[i]:=sqp[i]*sin(sangqp[i]);
end;

{ SET POWER LOSS }
for i:=1 to n do
begin
  plossp:=0;
  plosq:=0;
end;

{ FIND REAL POWER LOSS }
for i:=1 to n do
begin
  plossp:=plossp-spqr[i];
  plossp:=plossp+sqpr[i];
end;

{ FIND REACTIVE POWER LOSS }
for i:=1 to n do
begin
  plosq:=plosq-spqx[i];
  plosq:=plosq+sqpx[i];
end;
plossp:=plossp*sum[4];
plosq:=plosq*sum[4];

{ FIND DIFFERENT OF POWER LOSS }
dplossp:=abs(plosspb-plossp);
dplossq:=abs(plosqb-plosq);

{ FIND MINIMIZE COST FUNCTION }
totalcost:=0;costc:=0;costpl:=0;
for i:=1 to nc do
  costc:=costc+kvarcpu[i];
costc:=costc*kc*sum[4];
costpl:=kp*abs(plossp);
totalcost:=costc+costpl;

{ WRITE RESULT OF POWER LOSS }
pploss(n,plosspb,plossp,dplossp,plosqb,plosq,
       dplossq,costc,costpl,totalcost);
beep;

{*********************************************}
{* FIND LOCATION OF COMMERCIAL RATING CAPACITOR *}
{*********************************************}
{ SPECIFY COMMERCIAL RATING OF EACH CAPACITOR }
X11:clrscr;
writeln('*************************************************');
writeln('* FIND OPTIMUM LOCATION OF COMMERCIAL CAPACITOR *');
writeln('*************************************************');
writeln;
writeln('RATING OF OPTIMUM CAPACITOR SOMETIME DID NOT COMMERCIAL RATING');
writeln('THEREFORE YOU MUST SPECIFY RATING OF THOSE CAPACITORS IN THE');
writeln('COMMERCIAL RATING AND THEN THE RESULT WILL BE THE OPTIMUM');
writeln('LOCATION OF COMMERCIAL RATING OF CAPACITOR');
writeln;
for i:=1 to nc do
begin
  write('COMMERCIAL KVAR OF CAPACITOR (',I,') = ');
```

```
 readln(kvarc[i]);
 kvarcpu[i]:=kvarc[i]/sum[4];
 write('THIS CAPACITOR WILL BE INSTALLED AT BUS NUMBER = ');
 readln(busc[i]);writeln;
end;beep;

{*****************************************************}
{ SUBPROGRAM RADIAL LOAD FLOW AFTER INSTALL CAPACITORS }
{*****************************************************}
{ FIND RADIAL LOAD FLOW }
X9:cloadflow(n,nc,vbuspuold,angbusold,pup,puq,puir,puix,
             puilr,puilx,pur,pux,vrbuspuold,vxbuspuold,
             vrbuspunew,vxbuspunew,vbuspunew,angbusnew,
             kvarcpu,puirc,puixc,icrpu,icxpu,busc,
             vsubpu);
clrscr;
gotoxy(28,10);
writeln('Pass.......Load Flow');beep;

{***********************}
{ SUBPROGRAM FIND Fq(X) }
{***********************}
{ FIND Fq(X) }
 findfqx(n,angbusnew,puix,fqx);
 gotoxy(28,11);
 writeln('Pass.......Find Fq(x)');beep;
 for i:=1 to nc do
   buscold[i]:=busc[i];

{*************************************}
{ SUBPROGRAM A: FIND Icq  WHEN KNOWN Hi }
{*************************************}
for i:=1 to nc do
  opticq[i]:=kvarcpu[i]*vbuspunew[busc[i]]*cos(angbusnew[busc[i]]);
gotoxy(28,12);
writeln('Pass.......Subprogram A');beep;

{*************************************}
{ SUBPROGRAM B: FIND Hi WHEN KNOWN Icq }
{*************************************}
{ FIND 1/2*Icq + SIGMA Icq REST.}
for i:=1 to nc do
  ficq[i]:=opticq[i]/2;
for i:=2 to nc do
for j:=1 to i-1 do
  ficq[i]:=ficq[i]+opticq[j];

{ FIND Hi }
findh(n,nc,ficq,fqx,snormal,optbus);
gotoxy(28,13);
writeln('Pass.......Subprogram B');beep;

{*******************************}
{ COMPARE RESULT OF OPTIMUM BUS }
{*******************************}
{ COMPARE OPTIMUM BUS OF EACH CAPACITOR }
for i:=1 to nc do
begin
  if optbus[i]<>busc[i] then
  begin
```

```
    for i:=1 to nc do
      busc[i]:=optbus[i];
      goto X9;
  end;
end;

{ PRINT LOADFLOW }
ploadflow(n,vbuspunew,angbusnew);
writeln;
writeln('                      LoadFlow After Install Capacitor ');
writeln;
writeln('                       << Press Any Key to Continue >> ');
readln;beep;

{ PRINT LINE FLOW }
plineflow(n,puilr,puilx);
writeln;
writeln('                        LineFlow After Install Capacitor ');
writeln;
writeln('                         << Press Any Key to Continue >> ');
readln;beep;

{ PRINT Fq(x) }
pfqx(n,normal,snormal,fqx);beep;

{ PRINT RESULT }
presult(optbus,nc,kvarcpu,sum);
beep;

{****************************************}
{ FIND POWER LOSS AFTER INSTALL CAPACITORS }
{****************************************}
{ SET DIFFERENT VOLTAGE }
for i:=1 to n do
begin
  dvr[i]:=0;
  dvx[i]:=0;
end;

{ FIND DIFFERENT BUS VOLTAGE }
for i:=1 to n-1 do
begin
  dvr[i]:=vrbuspunew[i]-vrbuspunew[i+1];
  dvx[i]:=vxbuspunew[i]-vxbuspunew[i+1];
end;
dvr[n]:=vrbuspunew[n]-1;
dvx[n]:=vxbuspunew[n];
realpolar(n,dvr,dvx,dv,dang);

{ FIND POWER FLOW FROM P TO Q }
for i:=1 to n do
begin
  spq[i]:=vbuspunew[i]*dv[i]/zpu[i];
  sangpq[i]:=-angbusnew[i]+dang[i]-zang[i];
end;

{ FIND POWER FLOW FROM Q TO P }
for i:=1 to n do
begin
  dvr[i]:=-dvr[i];
```

```
  dvx[i]:=-dvx[i];
end;
realpolar(n,dvr,dvx,dv,dang);
sqp[n]:=dv[n]/zpu[n];
sangqp[n]:=dang[n]-zang[n];
for i:=n-1 downto 1 do
begin
  sqp[i]:=vbuspunew[i+1]*dv[i]/zpu[i];
  sangqp[i]:=-angbusnew[i+1]+dang[i]-zang[i];
end;

{ FIND REAL & REACTIVE POWER FLOW }
for i:=1 to n do
begin
  spqr[i]:=spq[i]*cos(sangpq[i]);
  spqx[i]:=spq[i]*sin(sangpq[i]);
  sqpr[i]:=sqp[i]*cos(sangqp[i]);
  sqpx[i]:=sqp[i]*sin(sangqp[i]);
end;

{ SET POWER LOSS }
for i:=1 to n do
begin
  plossp:=0;
  plosq:=0;
end;

{ FIND REAL POWER LOSS }
for i:=1 to n do
begin
  plossp:=plossp-spqr[i];
  plossp:=plossp+sqpr[i];
end;

{ FIND REACTIVE POWER LOSS }
for i:=1 to n do
begin
  plosq:=plosq-spqx[i];
  plosq:=plosq+sqpx[i];
end;
plossp:=plossp*sum[4];
plosq:=plosq*sum[4];

{ FIND DIFFERENT OF POWER LOSS }
dplossp:=abs(plosspb-plossp);
dplossq:=abs(plosqb-plosq);

{ FIND MINIMIZE COST FUNCTION }
totalcost:=0;costc:=0;costpl:=0;
for i:=1 to nc do
  costc:=costc+kvarcpu[i];
costc:=costc*kc*sum[4];
costpl:=kp*abs(plossp);
totalcost:=costc+costpl;

{ WRITE RESULT OF POWER LOSS }
ploss(n,plosspb,plossp,dplossp,plosqb,plosq,
      dplossq,costc,costpl,totalcost);
beep;
```

```
{ TRY ANOTHER SIZE }
X10:clrscr;
write('DO YOU WANT TO TRY ANOTHER COMMERCIAL RATING (Y/N)? ');
readln(yes);
if yes <> 'y' then
if yes <> 'n' then
if yes <> 'Y' then
if yes <> 'N' then
begin
  beep;
  writeln(' ERROR PLEASE TYPE Y/N ');
  goto X10;
end;
if (yes='y') or (yes='Y')then
  goto X11;
clrscr;
gotoxy(21,13);
writeln(' THIS IS THE END OF PROGRAM ');
beep;beep;beep;
END.
```

```
Program   Optimum_Size_and_Location_Of_Capacitors ;

          {---------------------------------------}
          {                                       }
          {           Program  Opt.Pas II         }
          {      Find Optimum Size and Location   }
          {                   Of                  }
          {     Capacitor in Distribution System  }
          {                                       }
          {---------------------------------------}

uses      dos,printer,crt,turbo3;
label     X1,X2,X3,X4,X5,X6,X7,X8,X9,X10;

type      matrix=array[0..30,0..30] of real;
          matric=array[0..30] of real;
          matint=array[0..30] of integer;

var       n,m,i,j,k,u:integer;{number of buses}
          vsub:real;          {substation voltage}
          dfrom:integer;      {data from}
          resist:matric;      {resistance in ohms per miles}
          react:matric;       {reactance in ohms per miles}
          phyl:matric;        {physical section lenght in miles}
          eqil:matric;        {length of equivalent uniform feeder}
          normal:matric;      {normalized length}
          kw:matric;          {kw load at bus i}
          kvar:matric;        {kvar load at bus i}
          maxkvar:matric;     {maximum kvar load}
          bkvarmax:integer;   {maximum kvar bus}
          sum:matric;         {summation of 1:physical length        }
                              {             2:uniform feeder in miles}
                              {             3:normalize feeder       }
                              {             4:kvar load              }
                              {             5:kw load                }
          snormal:matric;     {sum of normalized uniform feeder}
          yes:string[1];      {answer yes or no}
          filevar:file of real;{data files}
          name:string[15];    {file name}
          value:matric;
          rohm:matric;        {resistance in ohms}
          xohm:matric;        {reactance in ohms}
          h:matric;           {normalized bus distance from substation}
          rdist:matric;       {real distance from substation}
          basez:real;         {base impedance}
          basei:real;         {base current}
          aresist:real;       {all resistance in feeder}
          r:real;             {resistance in ohms/mile     }
                              {of equivalent uniform feeder}
          vsubpu:real;        {per unit voltage at substation}
          angsub:real;        {angle of substation}
          vbuspuold:matric;   {per unit bus voltage old}
```

```
angbusold:matric;    {angle of bus old}
vrbuspuold:matric;   {real part of per unit bus voltage old}
vxbuspuold:matric;   {imaginary part of per unit bus voltage old}
vbuspunew:matric;    {per unit bus voltage new}
angbusnew:matric;    {angle of bus new}
vrbuspunew:matric;   {real part of per unit bus voltage new}
vxbuspunew:matric;   {imaginary part of per unit bus voltage new}
pup:matric;          {per unit real power}
puq:matric;          {per unit reactive power}
pur:matric;          {per unit resistance}
pux:matric;          {per unit reactance}
puir:matric;         {per unit real load current}
puix:matric;         {per unit reactive load current}
puilr:matric;        {real part of per unit line current}
puilx:matric;        {imaginary part of per unit line current}
delta:matric;        {different value}
vdiffmax:real;       {maximum different voltage}
fx:matric;           {F(X)}
nc:integer;          {number of installed capacitors}
kvarcinput:matric;   {input kvar of each capacitor}
kvarcold:matric;     {kvar of each capacitor}
kvarcnew:matric;
kvarcpuold:matric;   {per unit kvar of each capacitor}
kvarcpunew:matric;
puirc:matric;        {real part of per unit      }
                     {load current with capacitor}
puixc:matric;        {imaginary part of per unit }
                     {load current with capacitor}
buscold:matint;      {bus that capacitor is installed}
buscnew:matint;      {bus that capacitor is installed new}
hc:matric;           {normalized distance of capacitor}
                     {from substation                  }
kp:real;             {annual cost per unit of power loss}
                     {in $/kw/yr                        }
ke:real;             {anual cost per unit of installed}
                     {capacitor in $/kvar three phases}
hcmatric:matrix;     {2d matric of normalized distance}
icq:matric;          {current of each capacitor}
int:matric;          {integrate fq(x)dx}
a:matrix;            {hcmatric+intmatric}
opticq:matric;       {optimum capacitor current}
ficq:matric;         {1/2Icq + sigma Icq rest}
optbus:matint;       {optimum bus distance of  }
                     {capacitor from substation}
icrpu:matric;        {per unit real current of each capacitor}
icxpu:matric;        {per unit imaginary current of each capacitor}
diffic:matric;       {difference between optimum capacitor      }
                     {current and the previous capacitor current}
difficmax:real;      {maximum difference of capacitor current}
sopticq:matric;      {sum of optimum Icq}
sopticd:matric;      {sum of optimum Icd}
plossq:real;         {power loss in q-axis}
plossd:real;         {power loss in d-axis}
plossbc:real;        {power loss before install capacitors}
p:real;              {power}
dvr:matric;          {different of real voltage p-q}
dvx:matric;          {different of imaginary voltage p-q}
dv:matric;           {different voltage}
dang:matric;         {different angle}
rpu:matric;          {resistance in per unit}
```

```
            xpu:matric;          {reactance in per unit}
            zpu:matric;          {per unit impedance}
            zang:matric;         {angle of impedance}
            spq:matric;          {power flow p-q}
            sangpq:matric;       {angle of power flow p-q}
            sqp:matric;          {power flow q-p}
            sangqp:matric;       {angle of power flow q-p}
            spqr:matric;         {power flow p-q real}
            spqx:matric;         {power flow p-q imaginary}
            sqpr:matric;         {power flow q-p real}
            sqpx:matric;         {power flow q-p imaginary}
            plossp:real;         {real power loss}
            plosq:real;          {reactive power loss}
            plosspb:real;        {real power loss before install capacitor}
            plosqb:real;         {reactive power loss befor install capacitor}
            dplossp:real;        {real power loss reduction}
            dplossq:real;        {reactive power loss reduction}
            plq:real;            {power due to q-axis}
            cost:real;           {minimizing cost function}
            captype:matint;      {type of capacitor}
            maxtype:integer;     {maximum type of capacitors}
            sum1,sum2,sum3:real;{sigma 1,2}
            suma,sumb,sumc,sumd,sume,sumf,sumg:matric;
            ist:matric;          {Ist(i)}
            kcs,kcf:real;        {annual cost per unit install fixed/switch}
                                 {capacitor in $/kvar 3 phase              }
            told:matric;         {inservice duration of capacitor}
            tnew:matric;
            lf:matric;           {load factor}
            fh:matric;           {optimum location from F(hi)}
            kf:real;             {annual fixed charge rate of each capacitor}
            hij:matrix;          {matric of Hij}
            bi:matric;           {matric of Bi}
            lp:matric;           {power loss reduction}
            slp:real;            {sum of power loss reduction}
            inttai:matric;       {integrate from 0 to Tai}
            le1:real;            {first term of LE}
            le2:real;            {second term of LE}
            le3:real;            {third term of LE}
            sle:real;            {sum of energy loss reduction}
            capcostfixed:real;   {cost of fixed capacitor}
            capcostswitched:real;{cost of switched capacitor}
            objfn:real;          {objective function}
            lpsave:real;         {save from LP}
            lesave:real;         {save form LE}
            capfixed:real;       {total fixed capacitor in per unit}
            capswitched:real;    {tatal switched capacitor in per unit}
            basepowerkw:real;    {value of I^2r}
            basekilowatthour:real;{value of I^2rt}

{ MAKING SOUND }
procedure beep;

begin
  sound(1000);
  delay(100);
  nosound;
end;

{ CHANGE FROM REAL TO POLAR }
```

```
procedure realpolar(n:integer;r,x:matric;var z,ang:matric);
var i:integer;

begin
for i:= 1 to n do
  begin
    z[i]:=sqrt(sqr(r[i])+sqr(x[i]));
    ang[i]:=arctan(x[i]/r[i]);
  end;
end;

{ CHANGE FROM POLAR TO REAL }
procedure polarreal(n:integer;z,ang:matric;var r,x:matric);
var i:integer;

begin
for i:=1 to n do
  begin
    r[i]:=z[i]*cos(ang[i]);
    x[i]:=z[i]*sin(ang[i]);
  end;
end;

{*************}
{* FIND F(X) *}
{*************}
procedure findfx(var n:integer;
                         var fx,kvar,sum:matric);
var i,j:integer;
    pla:matric;

begin
for i:=1 to n do
fx[i]:=0;
for i:=1 to n do
  pla[i]:=kvar[i]/sum[4];
for i:=1 to n do
for j:=i downto 1 do
  fx[i]:=fx[i]+pla[j];
end;

{ PRINT F(X) }
procedure pfx(var n:integer;
                  normal,snormal,fx:matric);
var i:integer;

begin
  clrscr;
  writeln('                                 *****************           ');
  writeln('                                 * TABLE OF F(X) *           ');
  writeln('                                 *****************           ');
  writeln;
  writeln('                    ¦------------------------¦--------------¦');
  writeln('                    ¦   NORMALIZED DISTANCE  ¦      F(X)    ¦');
  writeln('                    ¦------------------------¦--------------¦');
  for i:=1 to n-1 do
  writeln('                    ¦',snormal[i]:11:8,' - ',snormal[i+1]:10:8,
                          ' ¦',fx[i]:14:10,' ¦');
  writeln('                    ¦',snormal[n]:11:8,' - 0.00000000 ¦',
                             fx[n]:14:10,' ¦');
```

```
  writeln('                    |------------------------|--------------|');
  writeln;
  writeln('                         <<<  Press Any Key To Continue  >>>     ');
  readln;
end;

{*********************}
{ SUBPROGRAM FIND Icq }
{*********************}
procedure findicq(var nc:integer;
                  var a:matrix);
var k,l,m,t,p:integer;
    ri:real;

begin
  m:=nc+1;
  for i:=1 to nc do
  begin
    if a[i,i]=0 then
    begin
      for k:=i+1 to nc do
      begin
        if a[k,i] <> 0 then
        begin
          for l:=1 to m do
          begin
            ri:=a[i,l];
            a[i,l]:=a[k,l];
            a[k,l]:=ri;
          end;
        end;
      end;
    end;
  if a[i,i] <> 1 then
  begin
    ri:=1/a[i,i];
    for t:=i to m do
      a[i,t]:=a[i,t]*ri;
  end;
  for p:=1 to nc do
  begin
    if (p<>i) and (a[p,i]<>0.0) then
    begin
      ri:=a[p,i];
      for t:=i to m do
        a[p,t]:=a[p,t]-a[i,t]*ri;
    end;
  end;
  end;
end;

  {***********************************}
  {* FIND LOAD FACTOR WHEN SPECIFY Tj *}
  {***********************************}
  Procedure findlf(var told,lf:matric;
                   var maxtype:integer);
  var i:integer;

  begin
  told[0]:=1;
```

```
for i:=0 to maxtype do
begin
  if told[i]<=0.1875 then
    lf[i]:=-1.013333335*told[i]+1
  else
  begin
    lf[i]:=0.151875-0.27076923*(told[i]*told[i]-0.03515625);
    lf[i]:=lf[i]+0.72153846*(told[i]-0.1875);
    lf[i]:=lf[i]/told[i];
  end;
end;
gotoxy(30,10);write('Pass...Load Factor');beep;
end;

{*************************}
{* FIND OPTIMUM BANK SIZE *}
{*************************}
Procedure findici(var nc:integer;
                  var snormal,told,int,bi,fx,lf,
                      kvarcpunew:matric;
                  var hij,a:matrix;
                  var ke,kp,kcs,kcf,r:real;
                  var captype,buscold:matint);
var i,j:integer;

begin
{FIND Hij}
for i:=1 to nc do
begin
  for j:=1 to nc do
    hij[i,j]:=0;
end;
for i:=1 to nc do
begin
  for j:=1 to nc do
  begin
    if ((i>=j) and (captype[i]>=captype[j])) then
      hij[i,j]:=snormal[buscold[i]]*(kp+ke*told[captype[i]]*8760);
    if ((i>j) and (captype[i]<captype[j])) then
      hij[i,j]:=snormal[buscold[i]]*(kp+ke*told[captype[j]]*8760);
    if ((i<j) and (captype[i]>captype[j])) then
      hij[i,j]:=snormal[buscold[j]]*(kp+ke*told[captype[i]]*8760);
    if ((i<j) and (captype[i]<captype[j])) then
      hij[i,j]:=snormal[buscold[j]]*(kp+ke*told[captype[j]]*8760);
  end;
end;

{FIND Bi}
for i:=1 to nc do
begin
  int[i]:=0;
  bi[i]:=0;
end;

{find integrate F(x)dx from 0 to hi}
snormal[n+1]:=0;
for i:=1 to nc do
begin
  for j:=buscold[i] to n do
    int[i]:=int[i]+fx[j]*(snormal[j]-snormal[j+1]);
```

```
end;

for i:=1 to nc do
begin
if captype[i]=0 then
  bi[i]:=(kp+ke*told[captype[i]]*8760*lf[captype[i]])*int[i]-kcf/(2*r)
else
  bi[i]:=(kp+ke*told[captype[i]]*8760*lf[captype[i]])*int[i]-kcs/(2*r);
end;

{FIND Ici}
for i:=1 to nc do
begin
  for j:=1 to nc do
    a[i,j]:=hij[i,j];
end;
for i:=1 to nc do
  a[i,nc+1]:=bi[i];
findicq(nc,a);
for i:=1 to nc do
  kvarcpunew[i]:=a[i,nc+1];
gotoxy(30,11);write('Pass...Optimum Size');beep;
end;

{************************}
{* FIND OPTIMUM LOCATION *}
{************************}
Procedure findlocation(var nc:integer;
                       var sumd,sume,sumf,kvarcpunew,told,
                           lf,fh,fx:matric;
                       var sum1,kp,sum2,ke:real;
                       var captype,buscnew:matint);
var i,j:integer;

begin
{ SET THE VALUE OF SUMMATION }
for i:=1 to nc do
begin
  sumd[i]:=0;
  sume[i]:=0;
  sumf[i]:=0;
end;

{ SUMMATION OF THE FIRST TERM }
sum1:=0;
for i:=1 to nc do
begin
  for j:=i-1 downto 1 do
    sum1:=sum1+kvarcpunew[j];
  sum1:=sum1+kvarcpunew[i]/2;
  sumd[i]:=kp*sum1;
  sum1:=0;
end;

{ SUMMATION OF THE SECOND TERM }
sum1:=0; sum2:=0;
sume[1]:=ke*told[captype[1]]*8760*kvarcpunew[1]/2;
for i:=2 to nc do
begin
  for j:=i-1 downto 1 do
```

```
  begin
    if captype[i]>=captype[j] then
      suml:=suml+told[captype[i]]*8760*kvarcpunew[j]
    else
      suml:=suml+told[captype[j]]*8760*kvarcpunew[j];
  end;
  sum2:=suml+told[captype[i]]*8760*kvarcpunew[i]/2;
  sume[i]:=ke*sum2;
  suml:=0; sum2:=0;
end;

{ SUMMATION OF THE THIRD TERM }
for i:=1 to nc do
  sumf[i]:=kp+ke*told[captype[i]]*8760*lf[captype[i]];

{ FIND F(hi) }
for i:=1 to nc do
  fh[i]:=0;
for i:=1 to nc do
  fh[i]:=(sumd[i]+sume[i])/sumf[i];

{FIND Hi}
fx[0]:=0;
for i:=1 to nc do
  buscnew[i]:=0;
for i:=1 to nc do
begin
  for j:=0 to n-1 do
  begin
    if ((fh[i]>=fx[j])and(fh[i]<=fx[j+1])) then
      buscnew[i]:=j+1
  end;
end;
gotoxy(30,12);write('Pass...Optimum Location');beep;
end;

{***************}
{* FIND Isi(t) *}
{***************}
Procedure findinservice(var maxtype,nc,n:integer;
                        var suma,sumb,sumc,kvarcpunew,
                            snormal,fx,ist,tnew:matric;
                        var suml:real;
                        var captype,buscnew:matint);
var i,j,u,k:integer;

begin
{ SET THE VALUE OF SUMMATION }
for i:=1 to maxtype do
begin
  suma[i]:=0;
  sumb[i]:=0;
  sumc[i]:=0;
end;

{ SUMMATION OF THE FIRST TERM }
suml:=0;
for i:=1 to maxtype do
begin
  for j:=1 to nc do
```

```
  begin
    if captype[j]=i then
    begin
      u:=j-1;
      for k:=u downto 1 do
      begin
        if captype[k]<=i then
        sum1:=sum1+kvarcpunew[k];
      end;
    sum1:=sum1+kvarcpunew[j]/2;
    sum2:=sum1*snormal[buscnew[j]]*kvarcpunew[j];
    suma[i]:=suma[i]+sum1;
    sum1:=0;
    end;
  end;
end;

{ SUMMATION OF THE SECOND TERM }
sum1:=0;
for j:=1 to maxtype do
begin
  for i:=1 to nc do
  begin
    if captype[i]<j then
    begin
      for k:=i-1 downto 1 do
      begin
        if captype[k]=j then
        sum1:=sum1+kvarcpunew[k];
      end;
      sum1:=sum1*snormal[buscnew[i]]*kvarcpunew[i];
      sumb[j]:=sumb[j]+sum1;
      sum1:=0;
    end;
  end;
end;

{ SUMMATION OF THE THIRD TERM }
sum1:=0;
for i:=1 to maxtype do
begin
  for j:=1 to nc do
  begin
    if captype[j]=i then
    begin
      for k:=n downto buscnew[j] do
      sum1:=sum1+fx[k]*(snormal[k]-snormal[k+1]);
      sum1:=sum1*kvarcpunew[j];
    end;
  sumc[i]:=sumc[i]+sum1;
  sum1:=0;
  end;
end;

{ FIND Is(Ti) }
ist[0]:=0;
for i:=1 to maxtype do
  ist[i]:=0;
for i:=1 to maxtype do
  ist[i]:=(suma[i]+sumb[i])/sumc[i];
```

```
gotoxy(30,13);write('Pass...Switching Level');beep;

{ FIND INSERVICE DURATION Ti }
for i:=1 to maxtype do
begin
  if ((ist[i]<=1) and (ist[i]>=0.62)) then
    tnew[i]:=(ist[i]-1)/(-2.026666667);
  if (ist[i]<0.62) then
    tnew[i]:=(ist[i]-0.72153846)/(-0.54153846);
end;
gotoxy(30,14);write('Pass...Inservice Duration');beep;
end;

{*****************}
{WRITE THE RESULTS}
{*****************}
Procedure writeresult(var nc:integer;
                      var captype,buscnew:matint;
                      var kvarcpunew,sum,ist,tnew:matric);
var i:integer;

begin
gotoxy(30,15);write('Pass...Result Comparison');beep;
clrscr;
writeln('                          ********************          ');
writeln('                          * TABLE OF RESULT *          ');
writeln('                          ********************          ');
writeln;
writeln('        ¦-----¦------¦-----¦----------¦-------------¦------------¦');
writeln('        ¦  NO ¦ TYPE ¦ BUS ¦   KVAR   ¦  SWITCHING  ¦ IN-SERVICE ¦');
writeln('        ¦     ¦      ¦     ¦          ¦ LEVEL(KVAR) ¦DURATION(PU)¦');
writeln('        ¦-----¦------¦-----¦----------¦-------------¦------------¦');
for i:=1 to nc do
writeln('        ¦',i:3,'  ¦',captype[i]:4,'  ¦',buscnew[i]:3,'  ¦',
                kvarcpunew[i]*sum[4]:9:2,' ¦',ist[captype[i]]*sum[4]:11:2,
                '  ¦',tnew[captype[i]]:10:6,'  ¦');
writeln('        ¦-----¦------¦-----¦----------¦-------------¦------------¦');
writeln;
writeln('                     <<< Press Any Key To Continue >>>          ');
readln;
end;

{**************************}
{* FIND OBJECTIVE FUNCTION *}
{**************************}
{************}
{* FIND LPi *}
{************}
Procedure findobj(var nc,n,maxtype:integer;
                  var sumg,kvarcpunew,int,fx,snormal,lp,suma,
                      tnew,inttai,sumb,sumc:matric;
                  var r,slp,sum1,le1,sum2,sum3,le2,
                      le3,sle,capfixed,capswitched,objfn,lpsave,
                      lesave,capcostfixed,capcostswitched:real;
                  var buscnew,captype:matint);
var i,j,k:integer;

begin
for i:=1 to nc do
  sumg[i]:=0;
```

```
for i:=2 to nc do
begin
  for j:=i-1 downto 1 do
    sumg[i]:=sumg[i]+kvarcpunew[j];
end;
for i:=1 to nc do
  int[i]:=0;
for i:=1 to nc do
begin
  for j:=buscnew[i] to n do
    int[i]:=int[i]+(fx[j]-sumg[i])*(snormal[j]-snormal[j+1]);
end;

clrscr;
for i:=1 to nc do
  writeln('int f(x)-Ici=',int[i]);readln;

for i:=1 to nc do
  int[i]:=int[i]*2*r*kvarcpunew[i];
for i:=1 to nc do
  lp[i]:=0;
for i:=1 to nc do
  lp[i]:=int[i]-r*snormal[buscnew[i]]*sqr(kvarcpunew[i]);

for i:=1 to nc do
  writeln('LP=',lp[i]);readln;

slp:=0;
for i:=1 to nc do
  slp:=slp+lp[i];

writeln('sum lp = ',slp);readln;

{************}
{* FIND LEi *}
{************}
{ SUM OF THE FIRST TERM }
for i:=1 to nc do
  int[i]:=0;
snormal[n+1]:=0; sum1:=0; le1:=0;
for i:=1 to nc do
  suma[i]:=0;
for i:=1 to nc do
begin
  for j:=buscnew[i] to n do
    int[i]:=int[i]+fx[j]*(snormal[j]-snormal[j+1]);
end;
for i:=1 to nc do
  suma[i]:=2*r*tnew[captype[i]]*lf[captype[i]]*kvarcpunew[i]*int[i];
for i:=1 to nc do
  le1:=le1+suma[i];

clrscr;
for i:=1 to nc do
  writeln('suma = ',suma[i]);
writeln('le1 = ',le1);readln;

{ SUM OF THE SECOND TERM }
sum1:=0; sum2:=0; sum3:=0; le2:=0;
for i:=1 to nc do
```

```
begin
  for k:=1 to i-1 do
  begin
    if (captype[i]>=captype[k]) then
      sum1:=sum1+kvarcpunew[k];
  end;
  sum2:=tnew[captype[i]]*sum1;
  sum1:=0;
  for j:=captype[i]+1 to maxtype do
  begin
    for k:=1 to i-1 do
    begin
      if (captype[j]=captype[i]) then


        sum1:=sum1+kvarcpunew[k];
    end;
    sum3:=sum3+tnew[j]*sum1;
    sum1:=0;
  end;
  sumb[i]:=(sum2+sum3)*2*r*snormal[buscnew[i]]*kvarcpunew[i];
  sum1:=0; sum2:=0; sum3:=0;
end;
for i:=1 to nc do
  le2:=le2+sumb[i];

for i:=1 to nc do
  writeln('sumb = ',sumb[i]);
writeln('le2=',le2);readln;

{ SUM OF THE THIRD TERM }
le3:=0;
for i:=1 to nc do
  sumc[i]:=r*snormal[buscnew[i]]*tnew[captype[i]]*sqr(kvarcpunew[i]);
for i:=1 to nc do
  le3:=le3+sumc[i];

for i:=1 to nc do
  writeln('sumc=',sumc[i]);
writeln('le3=',le3); readln;

{ SUM OF LEi }
sle:=le1+le2+le3;

writeln('LE=',sle);
readln;

{*****************************}
{ CAPACITOR FIXED & SWITCHED }
{*****************************}
sum1:=0;sum2:=0;
for i:=1 to nc do
begin
  if (captype[i]=0) then
    sum1:=sum1+kvarcpunew[i]
  else
    sum2:=sum2+kvarcpunew[i];
end;
capfixed:=sum1;
```

```
  capswitched:=sum2;

  {*******************}
  { ANNUAL NET SAVING }
  {*******************}
  basepowerkw:=sqr(basei)*sum[2]/1000;
  basekilowatthour:=sqr(basei)*sum[2]*8760/1000;
  objfn:=0;
  lpsave:=kp*slp*3*basepowerkw;
  lesave:=ke*sle*3*basekilowatthour;
  capcostfixed:=capfixed*kcf*basepowerkw*(1+kf/100);
  capcostswitched:=capswitched*kcs*basepowerkw*(1+kf/100);
  objfn:=lpsave+lesave-capcostfixed-capcostswitched;

  {test}
  objfn:=kp*slp;writeln(objfn);
  objfn:=ke*sle*8760;writeln(objfn);
  objfn:=(kcf*capfixed+kcs*capswitched)*(1+kf/100);writeln(objfn);
  objfn:=kp*slp+ke*sle*8760+(1+kf/100)*(kcf*capfixed+kcs*capswitched);
  writeln('obj = ',objfn);

  end;

  {****************}
  { WRITE NET SAVE }
  {****************}
  Procedure writenetsave(var lpsave,lesave,capcostswitched,
                         capcostfixed,objfn:real);
  begin
  clrscr;
  writeln('                          **********************');
  writeln('                          * ANNUAL NET SAVING *');
  writeln('                          **********************');
  writeln;
  writeln('             ¦-------------------------------------¦----------¦');
  writeln('             ¦  SAVE FROM PEAK POWER LOSS REDUCTION  ¦',lpsave:8:0,'  ¦');
  writeln('             ¦  SAVE FROM ENERGY LOSS REDUCTION      ¦',lesave:8:0,'  ¦');
  writeln('             ¦  SWITCHED CAPACITOR COST              ¦',-capcostswitched:8:0,'  ¦');
  writeln('             ¦  FIXED CAPACITOR COST                 ¦',-capcostfixed:8:0,'  ¦');
  writeln('             ¦-------------------------------------¦----------¦');
  writeln('             ¦  ANNUAL NET SAVING ($)                ¦',objfn:8:0,'  ¦');
  writeln('             ¦-------------------------------------¦----------¦');
  writeln;
  writeln('                       <<< Press Any Key To Continue >>>     ');
  readln;
  end;

{*****************************************************************}

{*******************************}
{      MAIN PROGRAM OPT.PAS      }
{*******************************}
begin
  beep;
  clrscr;
  gotoxy(17,3); write('****************************************');
  gotoxy(17,4); write('*                                      *');
  gotoxy(17,5); write('*              Opto II                 *');
  gotoxy(17,6); write('*      Optimum Size And ...            *');
  gotoxy(17,7); write('*                Of                    *');
```

```
gotoxy(17,8); write('*   Capacitors In Distribution System   *');
gotoxy(17,9); write('*              Writen By                *');
gotoxy(17,10);write('*      Ass.Prof. Prasit Pittiyapat      *');
gotoxy(17,11);write('*                 And                   *');
gotoxy(17,12);write('*          Mr. Parames Chutima          *');
gotoxy(17,13);write('*       Chulalongkorn University        *');
gotoxy(17,14);write('*                                       *');
gotoxy(17,15);write('*****************************************');
gotoxy(17,20);write('   <<<  Press Any Key To Continue  >>> ');
readln;
clrscr;

{ GETTING INPUT DATA }
gotoxy(20,8);writeln('NUMBER OF BUSES = ');
gotoxy(38,8);readln(n);
gotoxy(20,9);writeln('VOLTAGE AT SUBSTATION IS = ');
gotoxy(52,9);writeln('KV');
gotoxy(48,9);readln(vsub);
gotoxy(20,10);writeln('DO YOU WANT TO GET DATA FROM ');
gotoxy(20,11);writeln('1) DATA FILES');
gotoxy(20,12);writeln('2) INPUT DATA FROM KEYBOARD');
gotoxy(20,13);writeln('(TYPE 1 OR 2) =======> ');
gotoxy(43,13);readln(dfrom);
if dfrom=2 then goto X1;
  clrscr;

{ GETTING INPUT DATA FROM DATA FILES }
{ RESIST.DAT & REAC.DAT & PHYL.DAT & KW.DAT & KVAR.DAT }
  write('DO YOU WANT TO CHANGE DATA IN DATA FILES (Y/N)? ');
  readln(yes);
  if yes = 'y' then

{ CHANGE DATA IN DATA FILES }
  begin
    writeln('DATA FILES THAT YOU CAN CHANGE IS ');
    writeln('1) RESIST.DAT');
    writeln('2) REACT.DAT');
    writeln('3) PHYL.DAT');
    writeln('4) KW.DAT');
    writeln('5) KVAR.DAT');
    write('TYPE A FILE NAME THAT YOU WANT TO CHANGE ===> ');
    X2: readln(name);
    assign(filevar,name);
    rewrite(filevar);
    for i:= 1 to n do
    begin
      write('INPUT ',name,' NUMBER (',i,') = ');
      readln(value[i]);
      write(filevar,value[i]);
    end;
    close(filevar);
    write('DO YOU WANT TO CHANGE ANOTHER FILES (Y/N)? ');
    readln(yes);
    if yes = 'y' then
    begin
      writeln('DATA FILES THAT YOU CAN CHANGE IS ');
      writeln('1) RESIST.DAT');
      writeln('2) REACT.DAT');
      writeln('3) PHYL.DAT');
      writeln('4) KW.DAT');
```

```
      writeln('5) KVAR.DAT');
      write('TYPE A FILE NAME THAT YOU WANT TO CHANGE ===> ');
      goto X2;
    end;
  end;

{ READ DATA FROM DATA FILES }
  assign(filevar,'resist.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,resist[i]);
  close(filevar);
  assign(filevar,'react.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,react[i]);
  close(filevar);
  assign(filevar,'phyl.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,phyl[i]);
  close(filevar);
  assign(filevar,'kw.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,kw[i]);
  close(filevar);
  assign(filevar,'kvar.dat');
  reset(filevar);
  for i:= 1 to filesize(filevar) do
  read(filevar,kvar[i]);
  close(filevar);
  goto X3;

{ INPUT DATA FROM KEYBOARD }
  X1: clrscr;
  writeln('THE FIRST BUS IS AT THE END OF FEEDER');
  for i:=1 to n do
  begin
    writeln('****  BUS NUMBER (',i,')',' ****');
    write('Resistance in Ohm/Miles = ');readln(resist[i]);
    write('Reactance in Ohm/Miles = ');readln(react[i]);
    write('Length in Miles = ');readln(phyl[i]);
    write('Three Phase Kilowatt Load = ');readln(kw[i]);
    write('Three Phase Kilovar Load = ');readln(kvar[i]);
    writeln;
  end;

{ FIND MAXIMUM KVAR }
X3: maxkvar[1]:=kvar[1];
bkvarmax:=1;                {find maximum kvar bus}
for i:=2 to n do
  if maxkvar[1]<kvar[i] then
  begin
    maxkvar[1]:=kvar[i];
    bkvarmax:=i;
  end;

{ SUM OF PHYSICAL LENGTH }
sum[1]:=0.0;
```

```
for i:=1 to n do
  sum[1]:=sum[1]+phyl[i];

{ SUM OF EQUIVALENT UNIFORM FEEDER IN MILES }
for i:=1 to n do
  eqil[i]:=resist[i]*phyl[i]/resist[bkvarmax];
sum[2]:=0.0;
for i:=1 to n do
  sum[2]:=sum[2]+eqil[i];

{ SUM OF NORMALIZED UNIFORM FEEDER }
for i:=1 to n do
  normal[i]:=eqil[i]/sum[2];
sum[3]:=0.0;
for i:=1 to n do
  sum[3]:=sum[3]+normal[i];

{ SUM OF KVAR }
sum[4]:=0.0;
for i:=1 to n do
  sum[4]:=sum[4]+kvar[i];

{ SUM OF KW }
sum[5]:=0.0;
for i:=1 to n do
  sum[5]:=sum[5]+kw[i];
clrscr;

{ WRITE CHARACTERISTIC RESULT FROM CALCULATION }
writeln('                        ************************');
writeln('                        * TABLE OF INPUT DATA *');
writeln('                        ************************');
writeln;
writeln(' ¦-----¦----------¦----------¦--------------¦--------------¦--------¦--------¦');
writeln(' ¦     ¦RESISTANCE¦ PHYSICAL ¦SECTION LENGTH¦NORMALIZED     ¦        ¦        ¦');
writeln(' ¦ BUS ¦OHMS/MILE ¦ SECTION  ¦OF EQUIVALENT ¦SECTION LENGTH¦   KW   ¦  KVAR  ¦');
writeln(' ¦ No. ¦          ¦ LENGTH   ¦UNIFORM FEEDER¦FOR EQUIVALENT¦  LOAD  ¦  LOAD  ¦');
writeln(' ¦     ¦          ¦ IN MILES ¦IN MILES      ¦UNIFORM FEEDER¦        ¦        ¦');
writeln(' ¦-----¦----------¦----------¦--------------¦--------------¦--------¦--------¦');

for i:=1 to n do
writeln(' ¦',i:3,'  ¦',resist[i]:9:5, ' ¦',phyl[i]:9:5,
        ' ¦',eqil[i]:11:5,'   ¦',normal[i]:11:5,
        '   ¦',kw[i]:8:1,'¦',kvar[i]:8:1,'¦');
writeln(' ¦-----¦----------¦----------¦--------------¦--------------¦--------¦--------¦');
writeln(' ¦     total      ¦',sum[1]:9:5,' ¦',sum[2]:11:5,
        '   ¦',sum[3]:11:5,'   ¦',sum[5]:8:1,'¦',sum[4]:8:1,'¦');
writeln(' ¦----------------¦----------¦--------------¦--------------¦--------¦--------¦');
writeln;
writeln('                        <<<  Press Any Key To Continue  >>>                  ');
readln;

{ FIND BUS TO BUS RESISTANCES }
for i:= 1 to n do
rohm[i]:=resist[i]*phyl[i];

{ FIND BUS TO BUS REACTANCES }
for i:= 1 to n do
xohm[i]:=react[i]*phyl[i];
```

```
{ WRITE R & X BETWEEN BUS }
clrscr;
writeln('                    ***********************************');
writeln('                    * VALUE OF R & X FROM BUS TO BUS *');
writeln('                    ***********************************');
writeln;
writeln('                   ¦---------¦------------¦------------¦');
writeln('                   ¦   BUS   ¦ RESISTANCE ¦ REACTANCE  ¦');
writeln('                   ¦         ¦   IN OHMS  ¦  IN OHMS   ¦');
writeln('                   ¦---------¦------------¦------------¦');
for i:= 1 to n do
writeln('                   ¦',i:3,' -',i+1:2,'  ¦',rohm[i]:10:5,'  ¦',
        xohm[i]:10:5,'  ¦');
writeln('                   ¦---------¦------------¦------------¦');
writeln;
writeln('                    <<<  Press Any Key To Continue  >>> ');
readln;

{ NORMALIZED EQUIVALENT DISTANCE H(X) }
for i:= 1 to n do
  h[i]:=0.0;
for i:= 1 to n do
for j:= i to n do
  h[i]:=h[i]+normal[j];

{ REAL DISTANCE OF BUS FROM SUBSTATION }
for i:= 1 to n do
  rdist[i]:=0.0;
for i:= 1 to n do
for j:= i to n do
  rdist[i]:=rdist[i]+phyl[j];

{ SUM OF NORMALIZED DISTANCES }
for i:=1 to n do
  snormal[i]:=0;
for i:=1 to n do
for j:=n downto i do
  snormal[i]:=snormal[i]+normal[j];

{ WRITE NORMALIZED & REAL DISTANCE }
clrscr;
writeln('                    *********************************');
writeln('                    * NORMALIZED & REAL  DISTANCE *');
writeln('                    *********************************');
writeln;
writeln('                   ¦-----¦------------¦------------¦');
writeln('                   ¦ BUS ¦ NORMALIZED ¦    REAL    ¦');
writeln('                   ¦     ¦  DISTANCE  ¦  DISTANCE  ¦');
writeln('                   ¦-----¦------------¦------------¦');
for i:= 1 to n do
writeln('                   ¦',i:3,'  ¦',snormal[i]:10:6,'  ¦',rdist[i]:10:4,'  ¦');
writeln('                   ¦-----¦------------¦------------¦');
writeln;
writeln('                  <<<  Press Any Key To Continue  >>>');
readln;

{ FIND BASE IN THIS PROGRAM }
basez:=sqr(vsub)*1000/sum[4];
basei:=sum[4]/(sqrt(3)*vsub);
aresist:=0.0;
```

```
for i:= 1 to n do
aresist:=aresist+resist[i]*phyl[i];
r:=aresist/sum[2];
clrscr;
writeln('{*****************************}');
writeln('{ BASE VALUE IN THIS PROBLEM }');
writeln('{*****************************}');
writeln;
writeln('Base Voltage (KV)       = ',vsub:12:4);
writeln('Base Power (KVA)        = ',sum[4]:12:4);
writeln('Base Current (AMP)      = ',basei:12:4);
writeln('Base Impedance (OHMS)   = ',basez:12:4);
writeln;
writeln('Resistance in Ohm/Miles Of Equivalent Uniform Feeder (r) = ',r:10:5);
writeln;
writeln('<<<  Press Any Key To Continue  >>>');
readln;

{ FIND PER UNIT  P,Q,R,X  FOR EACH BUS }
for i:=1 to n do
begin
  pup[i]:=kw[i]/sum[4];
  puq[i]:=kvar[i]/sum[4];
  pur[i]:=resist[i]*phyl[i]/basez;
  pux[i]:=react[i]*phyl[i]/basez;
end;

{ FIND PER UNIT RESISTANCES }
for i:=1 to n do
begin
  rpu[i]:=rohm[i]/basez;
  xpu[i]:=xohm[i]/basez;
end;
realpolar(n,rpu,xpu,zpu,zang);

{*************}
{* FIND F(X) *}
{*************}
findfx(n,fx,kvar,sum);
pfx(n,normal,snormal,fx);

{***********************************}
{ INPUT VALUE IN THE FIRST ITERATION }
{***********************************}
{ INPUT DATA IN THE FIRST ITERATION }
clrscr;
writeln('*****************************************');
writeln('* INPUT DATA IN THE FIRST ITERATION *');
writeln('*****************************************');
writeln;
write('NUMBER OF INSTALLED CAPACITORS = ');readln(nc);
writeln;
writeln('SPECIFY RATING & TYPE AND BUS OF CAPACITORS');
writeln('TYPE 0    *****>>  IS FIXED CAPACITOR');
writeln('TYPE > 0  *****>>  IS SWITCHING CAPACITOR');
write('INPUT MAXIMUM TYPE OF CAPACITORS = ');readln(maxtype);
for i:=1 to nc do
begin
  writeln;
  write('KVAR RATING OF CAPACITOR (',I,')   = ');readln(kvarc[i]);
```

```
  kvarcpuold[i]:=kvarc[i]/sum[4];
  write('INPUT TYPE OF THIS CAPACITOR    = ');readln(captype[i]);
  write('INPUT BUS THAT WILL INSTALL THIS CAPACITOR = ');
  readln(buscold[i]);
end;

 { INPUT INSERVICE TIME OF CAPACITORS }
writeln;
writeln('INSERVICE TIME(HOURS) OF CAPAICTORS TYPE 0 = 8760');
for i:=1 to maxtype do
begin
  write('INSERVICE TIME(HOURS) OF CAPACITOR TYPE (',I,') = ');
  readln(told[i]);
end;
for i:=1 to maxtype do
  told[i]:=told[i]/8760;


{ INPUT COST COEFFICIENT OF CAPACITORS }
writeln;
write('ANNUAL COST PER UNIT OF POWER LOSS IN $/KW/YR (Kp) = ');
{readln(kp);}kp:=120;
write('ANNUAL COST PER UNIT OF INSTALLED CAPACITOR IN $/KVAR 3 PHASE (Ke) = ');
{readln(ke);}ke:=15e-3;
write('ANNUAL COST PER UNIT OF INSTALLED FIXED CAPACITOR (Kcf) = ');
{readln(kcf);}kcf:=3.5;
write('ANNUAL COST PER UNIT OF INSTALLED SWITCH CAPACITOR (Kcs) = ');
{readln(kcs);}kcs:=6;
write('ANNUAL FIXED CHARGE RATE IN PERCENT OF CAPACITOR COST (Kf) =');
{readln(kf);}kf:=14.3;clrscr;

{************************************}
{* FIND LOAD FACTOR WHEN SPECIFY Tj *}
{************************************}
X6:findlf(told,lf,maxtype);

{**************************}
{* FIND OPTIMUM BANK SIZE *}
{**************************}
findici(nc,snormal,told,int,bi,fx,lf,kvarcpunew,
        hij,a,ke,kp,kcs,kcf,r,captype,buscold);

{**************************}
{* FIND OPTIMUM LOCATION *}
{**************************}
findlocation(nc,sumd,sume,sumf,kvarcpunew,told,lf,fh,
             fx,sum1,kp,sum2,ke,captype,buscnew);

{***************}
{* FIND Isi(t) *}
{***************}
findinservice(maxtype,nc,n,suma,sumb,sumc,kvarcpunew,
              snormal,fx,ist,tnew,sum1,captype,buscnew);

{**********************}
{* COMPARING RESULTS *}
{**********************}
tnew[0]:=1;
for i:=1 to nc do
begin
```

```
  if (abs(kvarcpunew[i]-kvarcpuold[i])>=0.0001) then
    goto X4;
  if (buscnew[i]<>buscold[i]) then
    goto X4;
end;
for i:=1 to maxtype do
begin
  if (abs(tnew[i]-told[i])>=0.001) then
    goto X4;
end;

{*****************}
{WRITE THE RESULTS}
{*****************}
writeresult(nc,captype,buscnew,kvarcpunew,sum,ist,tnew);


{test}
clrscr;
for i:=0 to maxtype do
  writeln('LF[',i,']=',lf[i]);
readln;clrscr;
for i:=1 to nc do
begin
  for j:=1 to nc do
    writeln('H[',i,',',j,']=',hij[i,j]);
end;readln;clrscr;
for i:=1 to nc do
  writeln('B[',i,']=',bi[i]);
readln;clrscr;
for i:=1 to nc do
  writeln('Ici=',kvarcpunew[i]);
for i:=0 to maxtype do
  writeln('Ti=',tnew[i]);readln;clrscr;



goto X5;



X4:{CHANGE NEW TO OLD ---> NEW ITERATION}
gotoxy(30,15);write('Not Pass...Result Comparison');beep;
for i:=1 to nc do
  kvarcpuold[i]:=kvarcpunew[i];
for i:=1 to nc do
  buscold[i]:=buscnew[i];
for i:=1 to maxtype do
  told[i]:=tnew[i];
clrscr;
goto X6;

{***************************}
{* FIND OBJECTIVE FUNCTION *}
{***************************}
X5:findobj(nc,n,maxtype,sumg,kvarcpunew,int,fx,snormal,lp,suma,
           tnew,inttai,sumb,sumc,r,slp,suml,lel,
           sum2,sum3,le2,le3,sle,capfixed,capswitched,objfn,
           lpsave,lesave,capcostfixed,capcostswitched,buscnew,
```

```
          captype);

{***************}
{ WRITE NET SAVE }
{***************}
writenetsave(lpsave,lesave,capcostswitched,capcostfixed,objfn);

{*********************************************************}
{ INPUT VALUE IN THE FIRST ITERATION OF PRACTICAL CAPACITOR}
{*********************************************************}
{ INPUT DATA IN THE FIRST ITERATION }
X10:clrscr;
writeln('*************************************');
writeln('* INPUT DATA IN THE FIRST ITERATION *');
writeln('*************************************');
writeln;
writeln('THE CALCULATED CAPACITORS SIZE MAY BE IMPRACTRICAL');
writeln('PLEASE SPECIFY THE PRACTRICAL SIZE OF CAPACITOR');
writeln('NUMBER OF INSTALLED CAPACITORS = ',NC);
writeln('SPECIFY KVAR RATING OF PRACTRICAL CAPACITORS');
writeln('TYPE = 0  *****>>  IS FIXED CAPACITOR');
writeln('TYPE > 0  *****>>  IS SWITCHING CAPACITOR');
writeln('THE MAXIMUM TYPE OF CAPACITORS = ',maxtype);
writeln;
for i:=1 to nc do
begin
  write('KVAR RATING OF CAPACITOR (',I,') = ');readln(kvarcinput[i]);
  kvarcpuold[i]:=kvarcinput[i]/sum[4];
  writeln('TYPE OF THIS CAPACITOR = ',captype[i]);
end;writeln;
writeln('Press Any Key To Continue');readln;
clrscr;
for i:=1 to nc do
  buscold[i]:=buscnew[i];
for i:=1 to maxtype do
  told[i]:=tnew[i];

{***********************************}
{* FIND LOAD FACTOR WHEN SPECIFY Tj *}
{***********************************}
X9:findlf(told,lf,maxtype);

{***************************}
{* FIND OPTIMUM BANK SIZE *}
{***************************}
for i:=1 to nc do
  kvarcpunew[i]:=kvarcpuold[i];
gotoxy(30,11);writeln('Pass...Optimum Size');

{************************}
{* FIND OPTIMUM LOCATION *}
{************************}
findlocation(nc,sumd,sume,sumf,kvarcpunew,told,lf,fh,
             fx,suml,kp,sum2,ke,captype,buscnew);

{***************}
{* FIND Isi(t) *}
{***************}
findinservice(maxtype,nc,n,suma,sumb,sumc,kvarcpunew,
              snormal,fx,ist,tnew,suml,captype,buscnew);
```

```
{*********************}
{* COMPARING RESULTS *}
{*********************}
tnew[0]:=1;
for i:=1 to nc do
begin
  if (abs(kvarcpunew[i]-kvarcpuold[i])>=0.0001) then
    goto X7;
  if (buscnew[i]<>buscold[i]) then
    goto X7;
end;
for i:=1 to maxtype do
begin
  if (abs(tnew[i]-told[i])>=0.001) then
    goto X7;
end;

{******************}
{WRITE THE RESULTS}
{******************}
writeresult(nc,captype,buscnew,kvarcpunew,sum,ist,tnew);



{test}
clrscr;
for i:=0 to maxtype do
  writeln('LF[',i,']=',lf[i]);
for i:=0 to maxtype do
  writeln('Ti=',tnew[i]);readln;clrscr;
for i:=1 to nc do
begin
  for j:=1 to nc do
    writeln('H[',i,',',j,']=',hij[i,j]);
end;readln;clrscr;
for i:=1 to nc do
  writeln('B[',i,']=',bi[i]);
readln;clrscr;
for i:=1 to nc do
  writeln('Ici=',kvarcpunew[i]);
readln;



goto X8;

X7:{CHANGE NEW TO OLD ---> NEW ITERATION}
gotoxy(30,15);write('Not Pass...Result Comparison');beep;
for i:=1 to nc do
  kvarcpuold[i]:=kvarcpunew[i];
for i:=1 to nc do
  buscold[i]:=buscnew[i];
```

```
  for i:=1 to maxtype do
    told[i]:=tnew[i];
  clrscr;
  goto X9;

  {**************************}
  {* FIND OBJECTIVE FUNCTION *}
  {**************************}
  X8:findobj(nc,n,maxtype,sumg,kvarcpunew,int,fx,snormal,lp,suma,
             tnew,inttai,sumb,sumc,r,slp,suml,le1,
             sum2,sum3,le2,le3,sle,capfixed,capswitched,objfn,
             lpsave,lesave,capcostfixed,capcostswitched,buscnew,
             captype);

  {****************}
  { WRITE NET SAVE }
  {****************}
  writenetsave(lpsave,lesave,capcostswitched,capcostfixed,objfn);

  clrscr;
  writeln('DO YOU WANT TO CONTINUE ANOTHER SIZE OF CAPACITOR ?');
  write('PLEASE TYPE (1) IF YOU WANT OR (2) IF YOU DO NOT WANT ==> ');
  readln(i);
  if i=1 then
    goto X10;
writeln;
writeln('<<< THIS IS THE END OF THIS PROGRAM >>>');
writeln('              GOOD BYE                  ');
sound(300);delay(1000);sound(600);delay(1000);sound(900);delay(1000);
nosound;
readln;
end.
```

ภาคผนวก ค

วิธีการใช้โปรแกรมสำหรับคะแปซิเตอร์แบบค่าคงที่

ในบทนี้จะขอกล่าวเกี่ยวกับวิธีการใช้โปรแกรม Optc1.Pas ซึ่งเป็นโปรแกรมที่พัฒนาขึ้นมาเพื่อใช้ในการวิเคราะห์หาขนาด และตำแหน่งติดตั้งที่เหมาะสมที่สุดสำหรับชันต์คะแปซิเตอร์แบบค่าคงที่เพียงชนิดเดียวเท่านั้น สำหรับวิธีการในการใช้โปรแกรมนี้จะมีดังนี้

เมื่อเปิดเครื่องคอมพิวเตอร์ขึ้นมาเราจะรัน (Run) โปรแกรมเทอร์โบปาสคาลก่อน หลังจากนั้นเราจะเรียกไฟล์ที่มีชื่อว่า Optc1.Pas เข้ามาในโหมดการเขียน ต่อจากนั้นเราจะรันโปรแกรมสำเร็จรูป Optc1.Pas จะทำให้เราได้รูปบนโมนิเตอร์ (Monitor) ออกมา

```
****************************************
*                                      *
*               Optc I                 *
*     Optimum Size And Location        *
*                 Of                   *
*  Capacitors In Distribution System   *
*              Writen By               *
*     Ass.Prof. Prasit Pittiyapat      *
*                 And                  *
*         Mr. Parames Chutima          *
*      Chulalongkorn University        *
*                                      *
****************************************
```

<<< Press Any Key To Continue >>>

หลังจากนั้นเครื่องจะให้เราตอบคำถามต่าง ๆ เราจะขอแทนการตอบโดยใช้สัญญลักษณ์วงกลมในที่นี้ ในตอนแรกเครื่องจะถามถึงจำนวนบัสที่มีอยู่ในระบบทั้งหมด ต่อมาจะถามถึงระดับแรงดันที่ใช้ในระบบ ต่อมาจะถามถึงวิธีการนำข้อมูลเข้าเครื่องว่าจะนำเข้าโดยการใช้ไฟล์ข้อมูล (Data File) ที่มีอยู่แล้ว หรือ จะดำเนินการนำเข้ากับข้อมูลใหม่ โดยเราจะเป็นผู้ใส่ตัวเลข 1 หรือ 2 เพื่อเลือกวิธีการนำเข้าข้อมูล

```
NUMBER OF BUSES = 9
VOLTAGE AT SUBSTATION IS =  23  KV
DO YOU WANT TO GET DATA FROM
1) DATA FILES
2) INPUT DATA FROM KEYBOARD
(TYPE 1 OR 2) ========> 1
```

เมื่อเราได้นำข้อมูลเข้าสู่เครื่องแล้วในกรณีที่เรานำข้อมูลเข้าโดยใช้ไฟล์ข้อมูล เราจะตอบคำถามต่อไปว่าเราจะมีการเปลี่ยนแปลงข้อมูลในไฟล์ข้อมูลหรือไม่ ให้เราตอบว่า Y หรือ N

```
DO YOU WANT TO CHANGE DATA IN DATA FILES (Y/N)? n
```

หลังจากนั้นเครื่องจะทำการพิมพ์ข้อมูลต่าง ๆ ที่เราได้กำหนดให้ในตอนแรกบนจอ โดยจะมีลำดับดังนี้คือ ตารางของข้อมูลที่นำเข้า

```
**********************
* TABLE OF INPUT DATA *
**********************
```

BUS No.	RESISTANCE OHMS/MILE	PHYSICAL SECTION LENGTH IN MILES	SECTION LENGTH OF EQUIVALENT UNIFORM FEEDER IN MILES	NORMALIZED SECTION LENGTH FOR EQUIVALENT UNIFORM FEEDER	KW LOAD	KVAR LOAD
1	1.37010	3.90000	6.19739	0.31623	1640.0	200.0
2	1.37010	3.50000	5.56176	0.28380	980.0	130.0
3	1.37010	1.50000	2.38361	0.12163	1150.0	60.0
4	0.86220	1.05000	1.05000	0.05358	780.0	110.0
5	0.86220	2.30000	2.30000	0.11736	1610.0	600.0
6	0.86220	0.81000	0.81000	0.04133	1598.0	1840.0
7	0.43900	1.70000	0.86558	0.04417	1790.0	446.0
8	0.28030	0.88000	0.28609	0.01460	980.0	340.0
9	0.19570	0.63000	0.14300	0.00730	1840.0	460.0
total		16.27000	19.59742	1.00000	12368.0	4186.0

```
<<<  Press Any Key To Continue  >>>
```

หลังจากนั้นเครื่องจะพิมพ์ข้อมูลของอิมพิแดนซ์ (R และ X) จากบัสหนึ่งไปยังอีกบัสหนึ่งดังนี้

```
**********************************
* VALUE OF R & X FROM BUS TO BUS *
**********************************
```

BUS	RESISTANCE IN OHMS	REACTANCE IN OHMS
1 - 2	5.34339	3.02640
2 - 3	4.79535	2.71600
3 - 4	2.05515	1.16400
4 - 5	0.90531	0.78860
5 - 6	1.98306	1.72760
6 - 7	0.69838	0.60840
7 - 8	0.74630	1.20500
8 - 9	0.24666	0.60510
9 -10	0.12329	0.41270

```
<<<  Press Any Key To Continue  >>>
```

จากนั้นเครื่องจะหาค่าของ Normalized distance โดยเปรียบเทียบกับระยะทางที่ใช้กันตามความเป็นจริง

```
*******************************
* NORMALIZED & REAL  DISTANCE *
*******************************
```

BUS	NORMALIZED DISTANCE	REAL DISTANCE
1	1.000000	16.2700
2	0.683765	12.3700
3	0.399964	8.8700
4	0.278336	7.3700
5	0.224757	6.3200
6	0.107395	4.0200
7	0.066063	3.2100
8	0.021895	1.5100
9	0.007297	0.6300

```
<<<  Press Any Key To Continue  >>>
```

หลังจากนั้นเครื่องจะพิมพ์ค่าฐาน (Base) ต่าง ๆ ที่ได้คำนวนได้

```
{**************************}
{ BASE VALUE IN THIS PROBLEM }
{**************************}

BASE VOLTAGE (KV)       =      23.0000
BASE POWER (KVA)        =    4186.0000
BASE CURRENT (AMP)      =     105.0777
BASE IMPEDANCE (OHMS)   =     126.3736

RESISTANCE IN OHM/MILES OF EQUIVALENT UNIFORM FEEDER (r) =     0.86220

<<<  Press Any Key To Continue  >>>
```

จากนั้นจะเป็นการคำนวนและแสดงผลของ $F_q(X)$

```
                  ******************
                  * TABLE OF Fq(X) *
                  ******************

¦-------------------------¦---------------¦
¦   NORMALIZED DISTANCE   ¦     Fq(X)     ¦
¦-------------------------¦---------------¦
¦ 1.00000000 - 0.68376501 ¦  0.0629762050 ¦
¦ 0.68376501 - 0.39996438 ¦  0.1038270891 ¦
¦ 0.39996438 - 0.27833554 ¦  0.1369508057 ¦
¦ 0.27833554 - 0.22475707 ¦  0.1742599373 ¦
¦ 0.22475707 - 0.10739469 ¦  0.3389422946 ¦
¦ 0.10739469 - 0.06606272 ¦  0.9334551533 ¦
¦ 0.06606272 - 0.02189485 ¦  1.1328591956 ¦
¦ 0.02189485 - 0.00729667 ¦  1.4089573526 ¦
¦ 0.00729667 - 0.00000000 ¦  2.2344966816 ¦
¦-------------------------¦---------------¦

    <<<  Press Any Key To Continue  >>>
```

ต่อจากนั้นเครื่องจะให้เราทำการกำหนดค่าเริ่มต้นให้กับคะแปซิเตอร์เพื่อใช้เป็นค่าในการดำเนินกระบวนการทำซ้ำ โดยที่เราจะเป็นผู้ใส่ค่าของจำนวนคะแปซิเตอร์ที่จะนำมาติดตั้ง ขนาดของคะแปซิเตอร์แต่ละตัว และบัสแต่ละบัสที่คะแปซิเตอร์เหล่านั้นจะติดตั้งลงไป รวมถึงค่าคงที่ k_p และ k_c ด้วย

```
**********************************************
* INPUT DATA OF CAPACITORS IN THE FIRST ITERATION *
**********************************************

EACH CAPACITOR WILL BE INSTALLED AT EACH BUS
THE FIRST BUS IS AT THE END OF THE FEEDER
NUMBER OF INSTALLED CAPACITORS = 3

KVAR RATING OF CAPACITOR (1) = 460
THIS CAPACITOR WILL BE INSTALLED AT BUS NUMBER = 1

KVAR RATING OF CAPACITOR (2) = 1070
THIS CAPACITOR WILL BE INSTALLED AT BUS NUMBER = 5

KVAR RATING OF CAPACITOR (3) = 2900
THIS CAPACITOR WILL BE INSTALLED AT BUS NUMBER = 6

DO YOU WANT TO CHANGE DATA OF CAPACITOR (Y/N)? n
INPUT CONSTANT COST COEFFICIENT OF CAPACITORS
ANNUAL COST PER UNIT OF POWER LOSS IN $/KW/YR (Kp) = 168
ANNUAL COST PER UNIT OF INSTALLED CAPACITOR IN $/KVAR 3 PHASE (Kc) = 4.9
```

จากนั้นเครื่องจะดำเนินการวิเคราะห์หาคำตอบของกระบวนการทำซ้ำ ซึ่งจะเป็นค่าของขนาด และตำแหน่งติดตั้งที่เหมาะสมที่สุดของคะแปซิเตอร์แบบค่าคงที่ เมื่อทำการวิเคราะห์จนเรียบร้อยแล้ว เครื่องจะพิมพ์ผลของโหลดโฟลว์ ไลน์โฟลว์ $F_q(x)$ และ Objective Function

```
*******************************
* OPTIMUM SIZE AND LOCATION   *
*******************************
```

BUS	OPTIMUM KVAR OF CAPACITOR
1	462.3317770600
5	1068.9591200000
6	2909.1259600000

```
<<<  Press Any Key To Continue  >>>
```

```
***************************
*  TABLE OF POWER LOSSES  *
***************************
```

	Before Installed Capacitor	After Installed Capacitor	Losses Reduction (kw/kvar)
P Loss	859.77	748.89	110.87
Objective Function ($)			147572.0

```
<<<  Press Any Key To Continue  >>>
```

ต่อจากนั้นถ้าเราต้องการที่จะวิเคราะห์ต่อสำหรับคะแปซิเตอร์ที่มีขนาดเท่ากับมาตรฐาน เราก็จะเป็นผู้ใส่ข้อมูลต่าง ๆ ของคะแปซิเตอร์ที่ได้มาตรฐาน และจะเป็นผู้กำหนดบัสต่าง ๆ ที่คะแปซิเตอร์แต่ละตัวจะถูกนำมาติดตั้ง ซึ่งค่านี้จะเป็นค่าเริ่มแรกในกระบวนการทำซ้ำ โดยที่คอมพิวเตอร์จะทำการหาตำแหน่งติดตั้งที่เหมาะสมสำหรับคะแปซิเตอร์ที่มีขนาดตามที่กำหนด

```
************************************************
* FIND OPTIMUM LOCATION OF COMMERCIAL CAPACITOR *
************************************************

RATING OF OPTIMUM CAPACITOR SOMETIME DID NOT COMMERCIAL RATING
THEREFORE YOU MUST SPECIFY RATING OF THOSE CAPACITORS IN THE
COMMERCIAL RATING AND THEN THE RESULT WILL BE THE OPTIMUM
LOCATION OF COMMERCIAL RATING OF CAPACITOR

COMMERCIAL KVAR OF CAPACITOR (1) = 470
THIS CAPACITOR WILL BE INSTALLED AT BUS NUMBER = 1

COMMERCIAL KVAR OF CAPACITOR (2) = 1070
THIS CAPACITOR WILL BE INSTALLED AT BUS NUMBER = 5

COMMERCIAL KVAR OF CAPACITOR (3) = 2910
THIS CAPACITOR WILL BE INSTALLED AT BUS NUMBER = 6
```

ต่อจากนั้นเครื่องจะดำเนินการหาค่าตอบของกระบวนการทำซ้ำ เพื่อหาตำแหน่งติดตั้งที่เหมาะสมของคะแปซิเตอร์ขนาดมาตรฐานที่กำหนดให้ แล้วมีข้อดีอีกประการหนึ่งก็คือในตัวโปรแกรมเองจะสามารถที่จะทำงานได้หลายรอบเพื่อที่จะทำให้เราสามารถวิเคราะห์หาการเปลี่ยนแปลงในตัวแปรของขนาดของคะแปซิเตอร์ ซึ่งในที่นี้ก็คือการทำ Sensivity Analysis นั่นเอง

```
*******************************
* OPTIMUM SIZE AND LOCATION   *
*******************************
```

BUS	OPTIMUM KVAR OF CAPACITOR
1	470.0000000000
5	1070.0000000000
6	2910.0000000000

<<< Press Any Key To Continue >>>

```
***************************
*  TABLE OF POWER LOSSES  *
***************************
```

	Before Installed Capacitor	After Installed Capacitor	Losses Reduction (kw/kvar)
P Loss	859.77	748.80	110.96
Objective Function ($)			147604.1

<<< Press Any Key To Continue >>>

ภาคผนวก ง

วิธีการใช้โปรแกรมสำหรับคะแปซิเตอร์แบบผสม

สำหรับวิธีการใช้งานโปรแกรมสำเร็จรูป Optc2.Pas เพื่อที่จะช่วยในการวิเคราะห์เกี่ยวกับการหาขนาด ตำแหน่งติดตั้ง และ ช่วงเวลาในการบริการ ที่เหมาะสมของคะแปซิเตอร์แบบผสมนั้น ในส่วนแรกซึ่งเป็นส่วนของการเริ่มต้นรันโปรแกรมจะมีวิธีการเดียวกันกับในภาคผนวก ค แต่จะมีข้อแตกต่างประการเดียวก็คือ แทนที่เราจะโหลดโปรแกรม Optc1.Pas เราจะโหลดโปรแกรม Optc2.Pas เข้ามาแทน หลังจากนั้นเราก็จะเริ่มต้นรันโปรแกรม Optc2.Pas สำหรับส่วนที่สองนั้นจะเป็นการใส่ข้อมูลต่าง ๆ ของลักษณะสมบัติต่าง ๆ ของสายป้อน รวมทั้งขนาดของโหลดที่เกิดขึ้นบนบัสแต่ละบัสด้วย ซึ่งวิธีการใส่ข้อมูลต่าง ๆ ก็จะเหมือนกันกับวิธีที่ใช้ในภาคผนวก ค สำหรับวิธีการใส่ข้อมูลในส่วนสุดท้ายนั้นจะแตกต่างกับภาคผนวก ค ดังต่อไปนี้

ในส่วนสุดท้ายของโปรแกรมนี้จะเป็นการใส่ค่าเริ่มต้นของกระบวนการทำซ้ำ เพื่อที่จะให้คอมพิวเตอร์ดำเนินการหาคำตอบที่เหมาะสมที่สุดออกมา โดยที่คอมพิวเตอร์จะมีการถามเพื่อให้เราใส่ค่าเริ่มต้นของ จำนวนคะแปซิเตอร์ที่จะติดตั้งบนบัสต่าง ๆ ซึ่งแน่นอนที่สุดจำนวนคะแปซิเตอร์ในระบบที่จะทำการวิเคราะห์จะต้องมีน้อยกว่าหรือเท่ากับจำนวนบัสที่มีอยู่ในระบบนั้น และอีกประการหนึ่งก็คือในที่นี้ขอกำหนดให้ว่าในแต่ละบัสจะมีคะแปซิเตอร์เพียงชนิดเดียวและตัวเดียวเท่านั้น หลังจากนั้นเครื่องจะให้เราทำการใส่ค่าขนาดของคะแปซิเตอร์ที่จะติดตั้งกับระบบ ต่อมาเครื่องจะถามต่ออีกว่าจะมีชนิดของคะแปซิเตอร์จำนวนกี่ชนิด ซึ่งชนิดของคะแปซิเตอร์ในที่นี้จะหมายความถึงว่า คะแปซิเตอร์ที่จะนำมาติดตั้งนั้นจะเป็นคะแปซิเตอร์แบบค่าคงที่หรือแบบสวิตชิง และสำหรับแบบสวิตชิงแล้วตัวไหนจะเป็นตัวที่จะมีการต่อเข้า/ปลดออกจากระบบก่อน โดยที่เราจะกำหนดให้ชนิด 0 เป็นคะแปซิเตอร์แบบค่าคงที่ สำหรับคะแปซิเตอร์ที่มีชนิดมากกว่า 0 จะถูกกำหนดให้เป็นคะแปซิเตอร์แบบสวิตชิงทั้งหมด โดยที่ชนิดที่น้อยกว่าจะมีการต่อเข้า/ปลดออกจากระบบก่อนชนิดที่มากกว่า ตัวอย่างเช่น ชนิดที่ 1 จะมีการต่อเข้า/ปลดออกจากระบบก่อนชนิดที่ 2, 3, ..., m ตามลำดับ โดยที่คะแปซิเตอร์ชนิด m เป็นชนิดสูงสุดของคะแปซิเตอร์ สุดท้ายเครื่องจะถามให้เราใส่ค่าเริ่มต้นของช่วงเวลาบริการของคะแปซิเตอร์แต่ละชนิด โดยที่เครื่องจะถูกกำหนดให้ในทันทีเลยว่า ช่วงเวลาในการบริการของคะแปซิเตอร์ชนิด 0 จะมีค่าเท่ากับ 1 เนื่องจากคะแปซิเตอร์ชนิด 0 เป็นคะแปซิเตอร์แบบค่าคงที่

หลังจากนั้นให้เราใส่ค่าช่วงเวลาบริการสำหรับคะแปซิเตอร์ชนิดต่าง ๆ จนกระทั้งครบ ๓ ค่า

```
**************************************
* INPUT DATA IN THE FIRST ITERATION *
**************************************

NUMBER OF INSTALLED CAPACITORS = 3

SPECIFY RATING & TYPE AND BUS OF CAPACITORS
TYPE 0    *****>>  IS FIXED CAPACITOR
TYPE > 0  *****>>  IS SWITCHING CAPACITOR
INPUT MAXIMUM TYPE OF CAPACITORS = 2

KVAR RATING OF CAPACITOR (1)    = 260
INPUT TYPE OF THIS CAPACITOR    = 1
INPUT BUS THAT WILL INSTALL THIS CAPACITOR = 1

KVAR RATING OF CAPACITOR (2)    = 680
INPUT TYPE OF THIS CAPACITOR    = 2
INPUT BUS THAT WILL INSTALL THIS CAPACITOR = 5

KVAR RATING OF CAPACITOR (3)    = 1500
INPUT TYPE OF THIS CAPACITOR    = 0
INPUT BUS THAT WILL INSTALL THIS CAPACITOR = 6

INSERVICE TIME(HOURS) OF CAPAICTORS TYPE 0 = 8760
INSERVICE TIME(HOURS) OF CAPACITOR TYPE (1) = 5000
INSERVICE TIME(HOURS) OF CAPACITOR TYPE (2) = 1700

ANNUAL COST PER UNIT OF POWER LOSS IN $/KW/YR (Kp) = 120
ANNUAL COST PER UNIT OF INSTALLED CAPACITOR IN $/KVAR 3 PHASE (Ke) = 15e-3
ANNUAL COST PER UNIT OF INSTALLED FIXED CAPACITOR (Kcf) = 3.5
ANNUAL COST PER UNIT OF INSTALLED SWITCH CAPACITOR (Kcs) = 6
```

ต่อจากนั้นเครื่องจะทำการหาค่าคำตอบที่เหมาะสมที่สุดสำหรับขนาด ตำแหน่งติดตั้ง และช่วงเวลาบริการของคะแปซิเตอร์แต่ละตัว และจะแสดงผลลัพธ์ที่ได้ออกมาโดยที่จะแสดงผลของแรงดัน และมุมเฟสของแต่ละบัส

```
************************
* RESULT OF LOAD FLOW *
************************
```

BUS	PU VOLT	PU ANG
1	0.85219076	-0.11770951
2	0.87170181	-0.10525552
3	0.90035123	-0.08975828
4	0.91781243	-0.08067408
5	0.92735056	-0.07321718
6	0.95519307	-0.05173693
7	0.96787722	-0.04316803
8	0.98666958	-0.02159951
9	0.99463955	-0.00960933

LoadFlow After Install Capacitor

ต่อจากนั้นคอมพิวเตอร์จะคำนวนหาค่าของการไหลของกระแส

```
*************
* LINE FLOW *
*************
```

BUS	PU Ir	PU Ix
1 - 2	0.45616632	-0.05684216
2 - 3	0.71949119	-0.12047445
3 - 4	1.02195448	-0.16367332
4 - 5	1.22200305	-0.20856848
5 - 6	1.63489620	-0.24892603
6 - 7	2.02798766	-0.38597576
7 - 8	2.46463160	-0.51501940
8 - 9	2.70007467	-0.60244497
9 - 10	3.14092178	-0.71716858

LineFlow After Install Capacitor

ต่อจากนั้นเครื่องจะทำการหาค่าของ Objective Function ให้ออกมาดังแสดงผลลัพธ์ดังนี้

```
******************
* TABLE OF RESULT *
******************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	1	261.07	1708.13	0.578869
2	2	5	652.31	2534.46	0.214348
3	0	6	1505.92	0.00	1.000000
ANNUAL NET SAVE $					6887

<<< Press Any Key To Continue >>>

ต่อจากนั้นจะเป็นการใส่ข้อมูลเริ่มต้นของคะแปซิเตอร์ที่มีขนาดเท่ากับมาตรฐาน โดยเราจะเป็นผู้ใส่ข้อมูลเหล่านี้ให้ ซึ่งคะแปซิเตอร์ที่มีขนาดมาตรฐานนั้นควรจะมีขนาดใกล้เคียงกันกับขนาดของคะแปซิเตอร์ที่มีค่าเหมาะสมที่สุดเพื่อที่จะทำให้ค่าของ Objective Function มี

ค่าเปลี่ยนแปลงไปไม่มาก

```
************************************
* INPUT DATA IN THE FIRST ITERATION *
************************************

THE CALCULATED CAPACITORS SIZE MAY BE IMPRACTRICAL
PLEASE SPECIFY THE PRACTRICAL SIZE OF CAPACITOR
NUMBER OF INSTALLED CAPACITORS = 3
SPECIFY KVAR RATING OF PRACTRICAL CAPACITORS
TYPE = 0  *****>>  IS FIXED CAPACITOR
TYPE > 0  *****>>  IS SWITCHING CAPACITOR
THE MAXIMUM TYPE OF CAPACITORS = 2

KVAR RATING OF CAPACITOR (1) = 270
TYPE OF THIS CAPACITOR = 1
KVAR RATING OF CAPACITOR (2) = 660
TYPE OF THIS CAPACITOR = 2
KVAR RATING OF CAPACITOR (3) = 1500
TYPE OF THIS CAPACITOR = 0

Press Any Key To Continue
```

หลังจากนั้นเครื่องจะทำการวิเคราะห์หาค่าคำตอบที่เหมาะสมที่สุดของตำแหน่งติดตั้งและช่วงเวลาในการบริการ สำหรับคะแปซิเตอร์แต่ละตัว โดยจะแสดงผลของโหลดโฟลว์ดังนี้

```
***********************
* RESULT OF LOAD FLOW *
***********************
```

BUS	PU VOLT	PU ANG
1	0.85241183	-0.11798286
2	0.87187432	-0.10544505
3	0.90047986	-0.08987706
4	0.91792216	-0.08076430
5	0.92744814	-0.07329592
6	0.95524099	-0.05176281
7	0.96791430	-0.04318324
8	0.98668627	-0.02160439
9	0.99464627	-0.00961094

```
LoadFlow After Install Capacitor

<< Press Any Key to Continue >>
```

ต่อจากนั้นจะแสดงให้เห็นถึงไลน์โฟลว์

```
*************
* LINE FLOW *
*************
```

BUS	PU Ir	PU Ix
1 - 2	0.45624951	-0.05511883
2 - 3	0.71951016	-0.11878541
3 - 4	1.02192506	-0.16201702
4 - 5	1.22194565	-0.20692486
5 - 6	1.63491916	-0.24558084
6 - 7	2.02791921	-0.38394913
7 - 8	2.46454445	-0.51299445
8 - 9	2.69998310	-0.60041969
9 - 10	3.14082705	-0.71514324

LineFlow After Install Capacitor

ต่อจากนั้นเครื่องจะทำการหาค่าของ Objective Function ของคะแปซิเตอร์ที่มีขนาดที่มีขนาดตามที่เราใส่ข้อมูลเข้าไปในตอนแรกตามต้องการ

```
******************
* TABLE OF RESULT *
******************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	1	270.00	1730.51	0.568998
2	2	5	660.00	2553.74	0.205840
3	0	6	1500.00	0.00	1.000000
ANNUAL NET SAVE $					6878

<<< Press Any Key To Continue >>>

ภาคผนวก จ

- ระบบไฟฟ้า 13 KV ที่มีจำนวนบัส 12 บัส
คิดตั้งคะแปซิเตอร์แบบค่าคงที่เพียงอย่างเดียว

ในตัวอย่างนี้จะแสดงให้เห็นถึงการคำนวนหาค่าที่เหมาะสมของขนาด และตำแหน่งติดตั้งของคะแปซิเตอร์กับระบบไฟฟ้า 13 kv และมีจำนวนบัสทั้งหมด 12 บัส ดังมีค่าพารามิเตอร์ต่าง ๆ ดังต่อไปนี้

Bus	R(Ohms)	X(Ohms)	Load(Kvar)
1	0.07550	0.15200	530
2	0.11300	0.22800	430
3	0.12600	0.25400	700
4	0.08190	0.16500	375
5	0.40700	0.21800	220
6	0.15200	0.15500	1085
7	0.10600	0.10800	0
8	0.16700	0.17100	1884
9	0.41000	0.41800	170
10	0.82000	0.83600	420
11	0.04480	0.08750	343
12	0.01040	0.00870	0

Bus	Normalized Distance	Real Distance
1	1.00000	12.00000
2	0.96996	11.00000
3	0.92500	10.00000
4	0.87488	9.00000
5	0.84229	8.00000
6	0.680379	7.00000
7	0.619908	6.00000
8	0.577737	5.00000
9	0.511299	4.00000
10	0.348186	3.00000
11	0.021961	2.00000
12	0.004137	1.00000

ในตัวอย่างนี้เราจะใช้ค่าฐาน (Base Value) ต่าง ๆ ดังต่อไปนี้

Base Voltage (KV) = 13

Base Power (KVA) = 6157

Base Current (Amp) = 273.44

Base Impedance (Ohm) = 27.44

ความต้านทานในหน่วย โอห์ม/ไมล์ ของสายป้อน Equivalent Uniform = 0.167

ค่าคำตอบของโหลดโฟล์วก่อนการติดตั้งคะแปซิเตอร์จะหาได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.92885	0.066736
2	0.92932	0.066499
3	0.93068	0.065786
4	0.93334	0.064382
5	0.93546	0.063267
6	0.93860	0.057071
7	0.94189	0.053682
8	0.94420	0.051332
9	0.94989	0.045581
10	0.96439	0.031300
11	0.99617	0.001938
12	0.99964	0.000383

ในที่นี้เราจะกำหนดค่าคงตัวต่าง ๆ ดังนี้

Annual Cost Per Unit of Power Loss ($/Kw/Yr) = 168

Annual Cost Per Unit of Installed Capacitor ($/KVar 3 ϕ) = 4.9

เราจะเริ่มทำการคำนวนหาค่าที่เหมาะสมที่สุดสำหรับคะแปซิเตอร์ โดยเราจะเริ่มต้นจากการติดตั้งคะแปซิเตอร์จำนวน 1 ตัว หลังจากนั้นจะเป็นการติดตั้ง 2 และ 3 ตัว ตามลำดับ ซึ่งเราจะขอกำหนดค่าเริ่มต้นในการทำ Iteration สำหรับคะแปซิเตอร์เพียงตัวเดียวดังต่อไปนี้

คะแปซิเตอร์ 1 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น = 500 KVar

ติดตั้งอยู่บนบัสที่ 1

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.97816	0.020191
2	0.97858	0.019964
3	0.97988	0.019307
4	0.98241	0.018031
5	0.98442	0.017019
6	0.98741	0.011419
7	0.98666	0.012197
8	0.98613	0.012740
9	0.98724	0.011680
10	0.99039	0.008664
11	0.99883	0.000638
12	0.99989	0.000122

ทำให้เราสามารถคำนวนหาค่าของ Fq(x) ได้ดังนี้

Bus	Fq(x)
1-2	0.0551831
2-3	0.1004458
3-4	0.1765356
4-5	0.2200718
5-6	0.2470074
6-7	0.4449850
7-8	0.4449850
8-9	0.7534085
9-10	0.7837293
10-11	0.8844021
11-12	1.9908039
12-Sub	1.9908039

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด = 4283.46 KVar

ติดตั้งที่บัสเลขที่ = 6

Objective Function = 27113.3

คะแปซิเตอร์ 2 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 3

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 2000 KVar

ติดตั้งอยู่บนบัสที่ 6

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.98124	0.017493
2	0.98170	0.017265
3	0.98300	0.016612
4	0.98273	0.016754
5	0.98292	0.016659
6	0.98347	0.015609
7	0.98488	0.014213
8	0.98586	0.013241
9	0.98700	0.012123
10	0.99022	0.008965
11	0.99881	0.000659
12	0.99989	0.000125

ทำให้เราสามารถคำนวนหาค่าของ Fq(x) ได้ดังนี้

Bus	Fq(x)
1-2	0.0625565
2-3	0.1139591
3-4	0.2008125
4-5	0.2469598
5-6	0.2741810
6-7	0.4173888
7-8	0.4173888
8-9	0.7098115
9-10	0.7385984
10-11	0.8344664
11-12	1.8888690
12-Sub	1.8888690

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 1895.65 KVar

ติดตั้งที่บัสเลขที่ = 3

ขนาดที่เหมาะสมที่สุด (2) = 2355.92 KVar

ติดตั้งที่บัสเลขที่ = 8

Objective Function = 25167.2

คะแปซิเตอร์ 3 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 2

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 4

ขนาดของคะแปซิเตอร์เริ่มต้น (3) = 2000 KVar

ติดตั้งอยู่บนบัสที่ 8

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.98137	0.017300
2	0.98182	0.017073
3	0.98180	0.017091
4	0.98284	0.016566
5	0.98293	0.016527
6	0.98333	0.015753
7	0.98463	0.014460
8	0.98554	0.013559
9	0.98671	0.012405
10	0.99002	0.009155
11	0.99879	0.000671
12	0.99989	0.000127

ทำให้เราสามารถคำนวนหาค่าของ Fq(x) ได้ดังนี้

Bus	Fq(x)
1-2	0.0636565
2-3	0.1159663
3-4	0.2010334
4-5	0.2480009
5-6	0.2756170
6-7	0.4184514
7-8	0.4184514
8-9	0.7059555
9-10	0.7342798
10-11	0.8287803
11-12	1.8713633
12-Sub	1.8713633

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 1002.47 KVar

ติดตั้งที่บัสเลขที่ = 2

ขนาดที่เหมาะสมที่สุด (2) = 1010.36 KVar

ติดตั้งที่บัสเลขที่ = 4

ขนาดที่เหมาะสมที่สุด (3) = 2206.26 KVar

ติดตั้งที่บัสเลขที่ = 8

Objective Function = 25008.6

ภาคผนวก ฉ

ระบบไฟฟ้า 12.5 KV ที่มีจำนวนบัส 14 บัส
ติดตั้งคะแปซิเตอร์แบบค่าคงที่เพียงอย่างเดียว

ในตัวอย่างนี้จะแสดงให้เห็นถึงการคำนวนหาค่าที่เหมาะสมของขนาด และตำแหน่งติดตั้งของคะแปซิเตอร์กับระบบไฟฟ้า 12.5 kv และมีจำนวนบัสทั้งหมด 14 บัส ดังมีค่าพารามิเตอร์ต่าง ๆ ดังต่อไปนี้

Bus	R(Ohms)	X(Ohms)	Load(Kvar)
1	0.18340	0.11340	600
2	0.14541	0.08991	300
3	0.34191	0.21141	200
4	0.17947	0.11097	1400
5	0.10710	0.07175	100
6	0.02785	0.08165	100
7	0.02876	0.01927	0
8	0.03060	0.02050	100
9	0.02142	0.01435	100
10	0.04284	0.02870	0
11	0.02754	0.01845	600
12	0.01224	0.00820	100
13	0.01224	0.00820	100
14	0.00918	0.00615	100

Bus	Normalized Distance	Real Distance
1	1.000000	7.53500
2	0.843243	6.13500
3	0.718956	5.02500
4	0.426715	2.41500
5	0.273317	1.04500
6	0.181775	0.69500
7	0.157975	0.60400
8	0.133389	0.51000
9	0.107234	0.41000
10	0.088926	0.34000
11	0.052309	0.20000
12	0.028770	0.11000
13	0.018308	0.07000
14	0.007847	0.03000

ในตัวอย่างนี้เราจะใช้ค่าฐาน (Base Value) ต่าง ๆ ดังต่อไปนี้

Base Voltage (KV) = 12.5

Base Power (KVA) = 3800

Base Current (Amp) = 175.51

Base Impedance (Ohm) = 41.11

ความต้านทานในหน่วย โอห์ม/ไมล์ ของสายป้อน Equivalent Uniform = 0.131

ค่าคำตอบของโหลดโฟล์วก่อนการติดตั้งคะแปซิเตอร์จะหาได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.99173	0.012846
2	0.99216	0.012125
3	0.99268	0.011274
4	0.99418	0.008835
5	0.99598	0.005935
6	0.99718	0.004142
7	0.99751	0.003658
8	0.99785	0.003159
9	0.99822	0.002608
10	0.99849	0.002210
11	0.99903	0.001413
12	0.99945	0.000795
13	0.99965	0.000513
14	0.99984	0.000223

ในที่นี้เราจะกำหนดค่าคงตัวต่าง ๆ ดังนี้

Annual Cost Per Unit of Power Loss ($/Kw/Yr) = 168

Annual Cost Per Unit of Installed Capacitor ($/KVar 3 ϕ) = 4.9

เราจะเริ่มทำการคำนวนหาค่าที่เหมาะสมที่สุดสำหรับคะแปซิเตอร์ โดยเราจะเริ่มต้นจากการติดตั้งคะแปซิเตอร์จำนวน 1 ตัว หลังจากนั้นจะเป็นการติดตั้ง 2 และ 3 ตัว ตามลำดับ ซึ่งเราจะขอกำหนดค่าเริ่มต้นในการทำ Iteration สำหรับคะแปซิเตอร์เพียงตัวเดียวดังต่อไปนี้

คะแปซิเตอร์ 1 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น = 500 KVar

ติดตั้งอยู่บนบัสที่ 1

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.99422	0.008928
2	0.99466	0.008217
3	0.99482	0.007945
4	0.99549	0.006864
5	0.99684	0.004677
6	0.99776	0.003306
7	0.99801	0.002933
8	0.99827	0.002547
9	0.99856	0.002117
10	0.99878	0.001802
11	0.99920	0.001173
12	0.99970	0.000663
13	0.99987	0.000429
14	0.99989	0.000187

ทำให้เราสามารถคำนวนหาค่าของ Fq(x) ได้ดังนี้

Bus	Fq(x)
1-2	0.0658064
2-3	0.1015398
3-4	0.1261720
4-5	0.3256313
5-6	0.3465112
6-7	0.3760185
7-8	0.3760185
8-9	0.4143046
9-10	0.4603524
10-11	0.4603524
11-12	0.9584172
12-13	1.1051345
13-14	1.3318632
14-Sub	1.8514178

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด = 616.810 KVar

ติดตั้งที่บัสเลขที่ = 2

Objective Function = 5737.40

คะแปซิเตอร์ 2 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 400 KVar

ติดตั้งอยู่บนบัสที่ 1

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 6

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.99447	0.008570
2	0.99462	0.008314
3	0.99492	0.007830
4	0.99589	0.006248
5	0.99711	0.004284
6	0.99794	0.003045
7	0.99817	0.002706
8	0.99840	0.002355
9	0.99867	0.001963
10	0.99886	0.001675
11	0.99925	0.001098
12	0.99957	0.000622
13	0.99972	0.000403
14	0.99988	0.000176

ทำให้เราสามารถคำนวนหาค่าของ Fq(x) ได้ดังนี้

Bus	Fq(x)
1-2	0.0652318
2-3	0.0988441
3-4	0.1226320
4-5	0.3310866
5-6	0.3527771
6-7	0.3832627
7-8	0.3832627
8-9	0.4226563
9-10	0.4699097
10-11	0.4699097
11-12	0.9762323
12-13	1.1251005
13-14	1.3549001
14-Sub	1.8808735

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 387.814 KVar

ติดตั้งที่บัสเลขที่ = 1

ขนาดที่เหมาะสมที่สุด (2) = 421.091 KVar

ติดตั้งที่บัสเลขที่ = 4

Objective Function = 6353.20

คะแปซิเตอร์ 3 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 300 KVar

ติดตั้งอยู่บนบัสที่ 1

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 600 KVar

ติดตั้งอยู่บนบัสที่ 7

ขนาดของคะแปซิเตอร์เริ่มต้น (3) = 800 KVar

ติดตั้งอยู่บนบัสที่ 8

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.99445	0.008591
2	0.99467	0.008237
3	0.99501	0.007675
4	0.99588	0.006270
5	0.99710	0.004298
6	0.99793	0.003055
7	0.99816	0.002714
8	0.99840	0.002362
9	0.99866	0.001969
10	0.99886	0.001679
11	0.99925	0.001101
12	0.99957	0.000624
13	0.99972	0.000404
14	0.99988	0.000176

ทำให้เราสามารถคำนวนหาค่าของ Fq(x) ได้ดังนี้

Bus	Fq(x)
1-2	0.0648930
2-3	0.0987243
3-4	0.1229224
4-5	0.3300873
5-6	0.3516486
6-7	0.3819587
7-8	0.3819587
8-9	0.4211319
9-10	0.4681259
10-11	0.4681259
11-12	0.9718503
12-13	1.1199650
13-14	1.3486120
14-Sub	1.8719712

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 304.274 KVar

ติดตั้งที่บัสเลขที่ = 1

ขนาดที่เหมาะสมที่สุด (2) = 164.061 KVar

ติดตั้งที่บัสเลขที่ = 3

ขนาดที่เหมาะสมที่สุด (3) = 333.836 KVar

ติดตั้งที่บัสเลขที่ = 4

Objective Function = 6287.3

ภาคผนวก ช

ระบบไฟฟ้า 13 KV ที่มีจำนวนบัส 12 บัส
ติดตั้งคะแปซิเตอร์แบบผสม

ในตัวอย่างนี้จะแสดงให้เห็นถึงการคำนวนหาค่าที่เหมาะสมของขนาด และตำแหน่งติดตั้งของคะแปซิเตอร์กับระบบไฟฟ้า 13 kv และมีจำนวนบัสทั้งหมด 12 บัส ดังมีค่าพารามิเตอร์ต่าง ๆ ดังต่อไปนี้

Bus	R(Ohms)	X(Ohms)	Load(Kvar)
1	0.07550	0.15200	530
2	0.11300	0.22800	430
3	0.12600	0.25400	700
4	0.08190	0.16500	375
5	0.40700	0.21800	220
6	0.15200	0.15500	1085
7	0.10600	0.10800	0
8	0.16700	0.17100	1884
9	0.41000	0.41800	170
10	0.82000	0.83600	420
11	0.04480	0.08750	343
12	0.01040	0.00870	0

Bus	Normalized Distance	Real Distance
1	1.00000	12.00000
2	0.96996	11.00000
3	0.92500	10.00000
4	0.87488	9.00000
5	0.84229	8.00000
6	0.680379	7.00000
7	0.619908	6.00000
8	0.577737	5.00000
9	0.511299	4.00000
10	0.348186	3.00000
11	0.021961	2.00000
12	0.004137	1.00000

ในตัวอย่างนี้เราจะใช้ค่าฐาน (Base Value) ต่าง ๆ ดังต่อไปนี้

Base Voltage (KV) = 13

Base Power (KVA) = 6157

Base Current (Amp) = 273.44

Base Impedance (Ohm) = 27.44

ความต้านทานในหน่วย โอห์ม/ไมล์ ของสายป้อน Equivalent Uniform = 0.167

ค่าคำตอบของโหลดโฟล์วก่อนการติดตั้งคะแปซิเตอร์จะหาได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.92885	0.066736
2	0.92932	0.066499
3	0.93068	0.065786
4	0.93334	0.064382
5	0.93546	0.063267
6	0.93860	0.057071
7	0.94189	0.053682
8	0.94420	0.051332
9	0.94989	0.045581
10	0.96439	0.031300
11	0.99617	0.001938
12	0.99964	0.000383

ในที่นี้เราจะกำหนดค่าคงตัวต่าง ๆ ดังนี้

Annual Cost Per Unit of Power Loss ($/Kw/Yr) = 120

Annual Cost Per Unit of Energy Loss Reduction ($/Kwh 3 ϕ) = 0.015

Annual Cost Per Unit of Installed Fixed Capacitor (Kcf) = 3.5

Annual Cost Per Unit of Installed Switched Capacitor (Kcs) = 6

เราจะเริ่มทำการคำนวนหาค่าที่เหมาะสมที่สุดสำหรับคะแปซิเตอร์ โดยเราจะเริ่มต้นจากการติดตั้งคะแปซิเตอร์จำนวน 1 ตัว หลังจากนั้นจะเป็นการติดตั้ง 2 ตัวตามลำดับ โดยที่เราจะแสดงให้เห็นถึงค่า Objective Function ที่แตกต่างกันเมื่อเราเปลี่ยนลำดับการสวิตชิ่งของคะแปซิเตอร์แต่ละตัว ดังนั้นเราจะสามารถหาค่าของ Fq(x) ที่ใช้งานได้ดังนี้

Bus	Fq(x)
1-2	0.0860808
2-3	0.1559200
3-4	0.2696118
4-5	0.3305181
5-6	0.3662497
6-7	0.5424719
7-8	0.5424719
8-9	0.8484651
9-10	0.8760760
10-11	0.9442910
11-12	1.0000000
12-Sub	1.0000000

คะแปซิเตอร์ 1 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น = 500 KVar

ติดตั้งอยู่บนบัสที่ 1 เป็นคะแปซิเตอร์แบบ 1

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.97231	0.023547
2	0.97278	0.023321
3	0.97409	0.022666
4	0.97566	0.021382
5	0.97468	0.021885
6	0.97500	0.023779
7	0.97522	0.023464
8	0.97753	0.023244
9	0.98365	0.020944
10	0.99814	0.014911
11	0.99982	0.001016
12	0.99982	0.000195

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด = 3150.52 KVar ติดตั้งที่บัสเลขที่ = 4

ระดับการสวิตชิ่ง = 2103.96 ช่วงเวลาบริการ = 0.701374

Objective Function = 27339

คะแปซิเตอร์ 2 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 500 KVar

ติดตั้งอยู่บนบัสที่ 1 เป็นคะแปซิเตอร์แบบ 0

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 3 เป็นคะแปซิเตอร์แบบ 1

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.98244	0.016219
2	0.98289	0.015994
3	0.98419	0.015348
4	0.98356	0.015675
5	0.98352	0.015700
6	0.98377	0.015233
7	0.98332	0.015677
8	0.98301	0.015987
9	0.98447	0.014547
10	0.98846	0.010603
11	0.99863	0.000760
12	0.99987	0.000144

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 2131.85 KVar ติดตั้งที่บัสเลขที่ = 3

ระดับการสวิตชิ่ง = 0 ช่วงเวลาบริการ = 1

ขนาดที่เหมาะสมที่สุด (2) = 1805.22 KVar ติดตั้งที่บัสเลขที่ = 6

ระดับการสวิตชิ่ง = 3529.43 ช่วงเวลาบริการ = 0.273848

Objective Function = 46635

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 500 KVar

ติดตั้งอยู่บนบัสที่ 1 เป็นคะแปซิเตอร์แบบ 1

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 3 เป็นคะแปซิเตอร์แบบ 0

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.98407	0.015630
2	0.98452	0.015406
3	0.98376	0.015795
4	0.98399	0.015687
5	0.98451	0.015431
6	0.98549	0.013566
7	0.98490	0.014155
8	0.98449	0.014568
9	0.98578	0.013296
10	0.98938	0.009758
11	0.99873	0.000708
12	0.99988	0.000134

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 1548.92 KVar ติดตั้งที่บัสเลขที่ = 2

ระดับการสวิตชิ่ง = 3672.68 ช่วงเวลาบริการ = 0.230885

ขนาดที่เหมาะสมที่สุด (2) = 2542.58 KVar ติดตั้งที่บัสเลขที่ = 6

ระดับการสวิตชิ่ง = 0 ช่วงเวลาบริการ = 1

Objective Function = 48993

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 500 KVar

ติดตั้งอยู่บนบัสที่ 1 เป็นคะแปซิเตอร์แบบ 1

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 3 เป็นคะแปซิเตอร์แบบ 1

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.97875	0.017419
2	0.97920	0.017191
3	0.98051	0.016541
4	0.98304	0.015274
5	0.98197	0.015829
6	0.98085	0.017969
7	0.98106	0.017759
8	0.98121	0.017612
9	0.98286	0.015981
10	0.98735	0.011570
11	0.99852	0.000817
12	0.99986	0.000156

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 3221.10 KVar ติดตั้งที่บัสเลขที่ = 4

ระดับการสวิตชิ่ง = 2348.61 ช่วงเวลาบริการ = 0.627997

ขนาดที่เหมาะสมที่สุด (2) = 538.79 KVar ติดตั้งที่บัสเลขที่ = 8

ระดับการสวิตชิ่ง = 2348.61 ช่วงเวลาบริการ = 0.627997

Objective Function = 29554

ภาคผนวก ช

ระบบไฟฟ้า 12.5 KV ที่มีจำนวนบัส 14 บัส
ติดตั้งคะแปซิเตอร์แบบผสม

ในตัวอย่างนี้จะแสดงให้เห็นถึงการคำนวนหาค่าที่เหมาะสมของขนาด และตำแหน่งติดตั้งของคะแปซิเตอร์กับระบบไฟฟ้า 12.5 kv และมีจำนวนบัสทั้งหมด 14 บัส ดังมีค่าพารามิเตอร์ต่าง ๆ ดังต่อไปนี้

Bus	R(Ohms)	X(Ohms)	Load(Kvar)
1	0.18340	0.11340	600
2	0.14541	0.08991	300
3	0.34191	0.21141	200
4	0.17947	0.11097	1400
5	0.10710	0.07175	100
6	0.02785	0.08165	100
7	0.02876	0.01927	0
8	0.03060	0.02050	100
9	0.02142	0.01435	100
10	0.04284	0.02870	0
11	0.02754	0.01845	600
12	0.01224	0.00820	100
13	0.01224	0.00820	100
14	0.00918	0.00615	100

Bus	Normalized Distance	Real Distance
1	1.000000	7.53500
2	0.843243	6.13500
3	0.718956	5.02500
4	0.426715	2.41500
5	0.273317	1.04500
6	0.181775	0.69500
7	0.157975	0.60400
8	0.133389	0.51000
9	0.107234	0.41000
10	0.088926	0.34000
11	0.052309	0.20000
12	0.028770	0.11000
13	0.018308	0.07000
14	0.007847	0.03000

ในตัวอย่างนี้เราจะใช้ค่าฐาน (Base Value) ต่าง ๆ ดังต่อไปนี้

Base Voltage (KV) = 12.5

Base Power (KVA) = 3800

Base Current (Amp) = 175.51

Base Impedance (Ohm) = 41.11

ความต้านทานในหน่วย โอห์ม/ไมล์ ของสายป้อน Equivalent Uniform = 0.131

ค่าคำตอบของโหลดโฟล์วก่อนการติดตั้งคะแปซิเตอร์จะหาได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.99173	0.012846
2	0.99216	0.012125
3	0.99268	0.011274
4	0.99418	0.008835
5	0.99598	0.005935
6	0.99718	0.004142
7	0.99751	0.003658
8	0.99785	0.003159
9	0.99822	0.002608
10	0.99849	0.002210
11	0.99903	0.001413
12	0.99945	0.000795
13	0.99965	0.000513
14	0.99984	0.000223

ในที่นี้เราจะกำหนดค่าคงตัวต่าง ๆ ดังนี้

Annual Cost Per Unit of Power Loss ($/Kw/Yr) = 120

Annual Cost Per Unit of Energy Loss Reduction ($/Kwh) =0.015

Annual Cost Per Unit of Installed Fixed Capacitor (Kcf) = 3.5

Annual Cost Per Unit of Installed Switched Capacitor (Kcs) = 6

เราจะเริ่มทำการคำนวนหาค่าที่เหมาะสมที่สุดสำหรับคะแปซิเตอร์ โดยเราจะเริ่มต้นจากการติดตั้งคะแปซิเตอร์จำนวน 1 ตัว หลังจากนั้นจะเป็นการติดตั้ง 2 ตัวตามลำดับ โดยที่เราจะแสดงให้เห็นถึงค่า Objective Function ที่แตกต่างกันเมื่อเราเปลี่ยนลำดับการสวิตชิ่งของคะแปซิเตอร์แต่ละตัว ดังนั้นเราจะสามารถหาค่าของ Fq(x) ที่ใช้งานได้ดังนี้

Bus	Fq(x)
1-2	0.1578947
2-3	0.2368421
3-4	0.2894736
4-5	0.6578947
5-6	0.6842102
6-7	0.7105263
7-8	0.7105263
8-9	0.7368421
9-10	0.7631578
10-11	0.7631578
11-12	0.9210526
12-13	0.9476368
13-14	0.9736842
14-Sub	1.0000000

คะแปซิเตอร์ 1 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น = 500 KVar

ติดตั้งอยู่บนบัสที่ 1 เป็นคะแปซิเตอร์แบบ 1

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.99588	0.006328
2	0.99631	0.005621
3	0.99624	0.005732
4	0.99635	0.005551
5	0.99741	0.003838
6	0.99814	0.002750
7	0.99834	0.002449
8	0.99855	0.002139
9	0.99879	0.001789
10	0.99896	0.001530
11	0.99931	0.001014
12	0.99961	0.000576
13	0.99974	0.000373
14	0.99986	0.000163

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด = 1025.08 KVar ติดตั้งที่บัสเลขที่ = 2

ระดับการสวิตชิ่ง = 0 ช่วงเวลาบริการ = 1

Objective Function = 1311

คะแปซิเตอร์ 2 ตัว

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 500 KVar

ติดตั้งอยู่บนบัสที่ 1 เป็นคะแปซิเตอร์แบบ 0

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 3 เป็นคะแปซิเตอร์แบบ 1

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.99650	0.005375
2	0.99693	0.004668
3	0.99685	0.004803
4	0.99692	0.004679
5	0.99779	0.003281
6	0.99839	0.002379
7	0.99856	0.002127
8	0.99874	0.001867
9	0.99894	0.001571
10	0.99096	0.001350
11	0.99938	0.000907
12	0.99965	0.000517
13	0.99977	0.000336
14	0.99990	0.000147

และจะสามารถหาค่าค่าตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 1049.55 KVar ติดตั้งที่บัสเลขที่ = 2

ระดับการสวิตชิ่ง = 0 ช่วงเวลาบริการ = 1

ขนาดที่เหมาะสมที่สุด (2) = 247.060 KVar ติดตั้งที่บัสเลขที่ = 4

ระดับการสวิตชิ่ง = 1621.30 ช่วงเวลาบริการ = 0.544522

Objective Function = 2654

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 500 KVar

ติดตั้งอยู่บนบัสที่ 1 เป็นคะแปซิเตอร์แบบ 1

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 3 เป็นคะแปซิเตอร์แบบ 0

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.99670	0.005047
2	0.99693	0.004679
3	0.99728	0.004106
4	0.99698	0.004586
5	0.99783	0.003221
6	0.99842	0.002340
7	0.99859	0.002093
8	0.99876	0.001838
9	0.99895	0.001547
10	0.99910	0.001330
11	0.99939	0.000896
12	0.99965	0.000511
13	0.99977	0.000332
14	0.99990	0.000145

และจะสามารถหาค่าค่าตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 288.49 KVar ติดตั้งที่บัสเลขที่ = 1

ระดับการสวิตชิ่ง = 1987.75 ช่วงเวลาบริการ = 0.366448

ขนาดที่เหมาะสมที่สุด (2) = 1036.69 KVar ติดตั้งที่บัสเลขที่ = 3

ระดับการสวิตชิ่ง = 0 ช่วงเวลาบริการ = 1

Objective Function = 2876

ขนาดของคะแปซิเตอร์เริ่มต้น (1) = 500 KVar

ติดตั้งอยู่บนบัสที่ 1 เป็นคะแปซิเตอร์แบบ 1

ขนาดของคะแปซิเตอร์เริ่มต้น (2) = 1000 KVar

ติดตั้งอยู่บนบัสที่ 3 เป็นคะแปซิเตอร์แบบ 1

หลังจากที่เราได้ทำการ Iteration จนเป็นผลสำเร็จแล้ว เราจะหาค่าของโหลดโฟล์วหลังจากติดตั้งคะแปซิเตอร์แล้วได้ดังนี้

Bus	p.u. Volt	p.u. angle
1	0.99638	0.005562
2	0.99681	0.004855
3	0.99671	0.004978
4	0.99683	0.004825
5	0.99772	0.003374
6	0.99835	0.002441
7	0.99853	0.002181
8	0.99871	0.001913
9	0.99891	0.001607
10	0.99906	0.001380
11	0.99937	0.000925
12	0.99964	0.000527
13	0.99976	0.000345
14	0.99989	0.000150

และจะสามารถหาค่าคำตอบของขนาดและตำแหน่งที่เหมาะสมที่สุดได้คือ

ขนาดที่เหมาะสมที่สุด (1) = 1036.76 KVar ติดตั้งที่บัสเลขที่ = 2

ระดับการสวิตชิ่ง = 1105.70 ช่วงเวลาบริการ = 0.795076

ขนาดที่เหมาะสมที่สุด (2) = 214.38 KVar ติดตั้งที่บัสเลขที่ = 4

ระดับการสวิตชิ่ง = 1105.70 ช่วงเวลาบริการ = 0.795076

Objective Function = 1403

ภาคผนวก ฉ

การเปรียบเทียบค่าฟังก์ชันวัตถุประสงค์
สำหรับคะแปซิเตอร์แบบค่าคงที่ซึ่งใช้กับระบบ 9 บัส 13 kV

```
*******************************
* OPTIMUM SIZE AND LOCATION   *
*******************************
```

BUS	OPTIMUM KVAR OF CAPACITOR
6	4112.5107808000

<<< Press Any Key To Continue >>>

```
*************************
*  TABLE OF POWER LOSSES *
*************************
```

	Before Installed Capacitor	After Installed Capacitor	Losses Reduction (kw/kvar)
P Loss	859.77	768.17	91.60
Objective Function ($)			149204.0

<<< Press Any Key To Continue >>>

****
* OPTIMUM SIZE AND LOCATION *
****

BUS	OPTIMUM KVAR OF CAPACITOR
2	771.6500264400
6	3637.9919645000

<<< Press Any Key To Continue >>>

****
* TABLE OF POWER LOSSES *
****

	Before Installed Capacitor	After Installed Capacitor	Losses Reduction (kw/kvar)
P Loss	859.77	753.65	106.11
Objective Function ($)			148221.0

<<< Press Any Key To Continue >>>

****
* OPTIMUM SIZE AND LOCATION *
****

BUS	OPTIMUM KVAR OF CAPACITOR
1	462.3266040700
5	1069.0239829000
6	2908.9637631000

<<< Press Any Key To Continue >>>

****
* TABLE OF POWER LOSSES *
****

	Before Installed Capacitor	After Installed Capacitor	Losses Reduction (kw/kvar)
P Loss	859.77	748.89	110.87
Objective Function ($)			147571.6

<<< Press Any Key To Continue >>>

****
* OPTIMUM SIZE AND LOCATION *
****

BUS	OPTIMUM KVAR OF CAPACITOR
1	300.0920797700
2	268.6839748400
5	968.3235849600
6	2920.2410833000

<<< Press Any Key To Continue >>>

****
* TABLE OF POWER LOSSES *
****

	Before Installed Capacitor	After Installed Capacitor	Losses Reduction (kw/kvar)
P Loss	859.77	748.12	111.64
Objective Function ($)			147525.3

<<< Press Any Key To Continue >>>

****
* OPTIMUM SIZE AND LOCATION *
****

BUS	OPTIMUM KVAR OF CAPACITOR
1	300.4567785900
2	189.8065677700
3	200.5307302700
5	848.5615920900
6	2924.5798825000

<<< Press Any Key To Continue >>>

****
* TABLE OF POWER LOSSES *
****

	Before Installed Capacitor	After Installed Capacitor	Losses Reduction (kw/kvar)
P Loss	859.77	747.85	111.92
Objective Function ($)			147512.2

<<< Press Any Key To Continue >>>

****
* OPTIMUM SIZE AND LOCATION *
****

BUS	OPTIMUM KVAR OF CAPACITOR
1	300.4318004700
2	189.8076213600
3	148.7930096100
4	165.9273053500
5	734.8417510200
6	2927.8126188000

<<< Press Any Key To Continue >>>

****
* TABLE OF POWER LOSSES *
****

	Before Installed Capacitor	After Installed Capacitor	Losses Reduction (kw/kvar)
P Loss	859.77	747.74	112.02
Objective Function ($)			147512.1

<<< Press Any Key To Continue >>>

****
* OPTIMUM SIZE AND LOCATION *
****

BUS	OPTIMUM KVAR OF CAPACITOR
1	300.1876658600
2	189.6701769000
3	148.6667609100
4	165.8398049500
5	734.6193747600
6	2590.7929594000
7	530.9145506300

<<< Press Any Key To Continue >>>

****
* TABLE OF POWER LOSSES *
****

	Before Installed Capacitor	After Installed Capacitor	Losses Reduction (kw/kvar)
P Loss	859.77	746.32	113.44
Objective Function ($)			148219.9

<<< Press Any Key To Continue >>>

ภาคผนวก ญ

การเปรียบเทียบค่าฟังก์ชันวัตถุประสงค์
สำหรับคะแปซิเตอร์แบบค่าผสมซึ่งใช้กับระบบ 9 บัส 13 kV

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	5	1442.43	1398.76	0.715343
ANNUAL NET SAVE $					2920

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	0	1	271.48	0.00	1.000000
2	1	5	1232.15	1721.37	0.573031
		ANNUAL NET SAVE $			3616

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	2	469.62	2259.75	0.335531
2	0	6	1787.30	0.00	1.000000
		ANNUAL NET SAVE $			6195

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	0	1	239.18	0.00	1.000000
2	1	4	184.41	754.26	0.999655
3	2	5	511.68	1317.72	0.751095
4	3	6	1241.04	1964.62	0.465726
ANNUAL NET SAVE $					4891

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	0	1	239.13	0.00	1.000000
2	1	4	178.60	747.53	1.002623
3	3	5	666.30	2516.06	0.222463
4	2	6	1143.57	1249.54	0.781172
ANNUAL NET SAVE $					6334

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	0	1	194.03	0.00	1.000000
2	2	2	137.85	2038.83	0.432986
3	1	5	615.10	972.77	0.903262
4	3	6	1231.02	1973.05	0.462004
ANNUAL NET SAVE $					4767

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	0	1	199.89	0.00	1.000000
2	2	2	141.42	1985.91	0.456332
3	3	5	724.14	2440.71	0.255704
4	1	6	1161.66	985.88	0.897480
ANNUAL NET SAVE $					5995

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	0	1	173.32	0.00	1.000000
2	3	2	232.77	2918.05	0.149459
3	1	5	561.29	880.44	0.943992
4	2	6	1219.31	1697.50	0.583558
ANNUAL NET SAVE $					7042

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	0	1	194.64	0.00	1.000000
2	3	2	185.95	3026.19	0.136712
3	2	5	686.51	2088.22	0.411201
4	1	6	1127.64	957.76	0.909884
ANNUAL NET SAVE $					7063

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	1	234.35	1812.73	0.532729
2	0	5	858.65	0.00	1.000000
3	2	6	582.48	1747.97	0.561295
4	3	6	583.42	2484.11	0.236559
ANNUAL NET SAVE $					6605

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	1	234.36	1812.85	0.532675
2	0	5	858.71	0.00	1.000000
3	3	6	582.66	2484.76	0.236274
4	2	6	583.29	1748.59	0.561024
ANNUAL NET SAVE $					6607

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	1	164.25	1601.41	0.625950
2	2	2	130.76	2236.59	0.345750
3	0	5	853.72	0.00	1.000000
4	3	6	1056.38	2117.56	0.398255
ANNUAL NET SAVE $					5261

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	1	165.38	1424.70	0.703903
2	2	2	178.71	2149.18	0.384306
3	3	5	580.48	2620.21	0.184566
4	0	6	1499.88	0.00	1.000000
ANNUAL NET SAVE $					7168

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	1	135.72	1464.26	0.686450
2	3	2	215.39	2958.35	0.144709
3	0	5	812.77	0.00	1.000000
4	2	6	1046.26	1858.32	0.512615
ANNUAL NET SAVE $					7291

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	1	1	167.01	1416.97	0.707309
2	3	2	204.64	2983.16	0.141783
3	2	5	543.92	2222.88	0.351795
4	0	6	1479.99	0.00	1.000000
ANNUAL NET SAVE $					7646

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	2	1	131.14	2092.93	0.409121
2	0	2	234.59	0.00	1.000000
3	1	5	587.87	1025.05	0.880202
4	3	6	1225.33	1977.84	0.459892
ANNUAL NET SAVE $					4731

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	2	1	131.63	2099.24	0.406337
2	0	2	246.52	0.00	1.000000
3	3	5	692.67	2482.07	0.237458
4	1	6	1162.14	1045.06	0.871375
ANNUAL NET SAVE $					6093

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	2	1	172.20	2224.07	0.351273
2	3	2	180.68	3038.32	0.135281
3	0	5	820.49	0.00	1.000000
4	1	6	1024.54	1682.98	0.589967
		ANNUAL NET SAVE $			7461

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	2	1	171.77	2093.57	0.408839
2	3	2	212.63	2964.72	0.143957
3	1	5	502.81	1862.34	0.510845
4	0	6	1483.46	0.00	1.000000
		ANNUAL NET SAVE $			7807

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	2	1	175.16	2105.15	0.403732
2	1	2	118.84	1487.14	0.676355
3	0	5	851.38	0.00	1.000000
4	3	6	1059.32	2115.20	0.399296
ANNUAL NET SAVE $					5269

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	2	1	179.74	2118.59	0.397801
2	1	2	163.63	1389.04	0.719631
3	3	5	582.65	2617.98	0.184829
4	0	6	1496.70	0.00	1.000000
ANNUAL NET SAVE $					7161

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	3	1	194.27	2919.63	0.149273
2	0	2	215.47	0.00	1.000000
3	1	5	550.53	951.76	0.912531
4	2	6	1223.01	1739.49	0.565039
ANNUAL NET SAVE $					7058

<<< Press Any Key To Continue >>>

****
* TABLE OF RESULT *
****

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	3	1	161.70	3020.67	0.137362
2	0	2	241.00	0.00	1.000000
3	2	5	666.80	2161.01	0.379088
4	1	6	1129.79	1017.66	0.883461
ANNUAL NET SAVE $					7029

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	3	1	221.44	2834.57	0.159298
2	1	2	111.52	1431.98	0.700690
3	0	5	824.23	0.00	1.000000
4	2	6	1050.10	1844.66	0.518644
ANNUAL NET SAVE $					7134

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	3	1	196.73	2912.00	0.150172
2	1	2	173.63	1398.24	0.715573
3	2	5	544.61	2234.76	0.346556
4	0	6	1478.21	0.00	1.000000
ANNUAL NET SAVE $					7543

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	3	1	201.84	2896.00	0.152057
2	2	2	144.70	2212.41	0.356415
3	0	5	824.75	0.00	1.000000
4	1	6	1026.86	1689.82	0.586946
ANNUAL NET SAVE $					7285

<<< Press Any Key To Continue >>>

```
********************
* TABLE OF RESULT *
********************
```

NO	TYPE	BUS	KVAR	SWITCHING LEVEL(KVAR)	IN-SERVICE DURATION(PU)
1	3	1	205.34	2885.08	0.153345
2	2	2	171.15	2131.27	0.392209
3	1	5	509.72	1869.64	0.507622
4	0	6	1484.12	0.00	1.000000
ANNUAL NET SAVE $					7689

<<< Press Any Key To Continue >>>

ประวัติผู้เขียน

ผู้เรียบเรียงวิทยานิพนธ์ฉบับนี้ชื่อ นายปารเมศ ชุติมา เกิดเมื่อวันที่ 28 กุมภาพันธ์ พ.ศ.2506 เป็นคนจังหวัดกรุงเทพฯ จบการศึกษาในระดับปริญญาตรี (เกียรตินิยม) สาขาวิศวกรรมไฟฟ้า จากคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2529 หลังจากจบการศึกษาในระดับปริญญาตรีแล้วก็ได้เข้าศึกษาต่อในระดับปริญญาโทที่จุฬาลงกรณ์มหาวิทยาลัย ทันทีในสาขาวิศวกรรมไฟฟ้า แผนกพลังงานไฟฟ้า แต่ยังไม่ทันจะสำเร็จก็ได้รับทุนรัฐบาลญี่ปุ่นให้เข้าไปศึกษาต่อในระดับปริญญาโท สาขาวิชา Industrial Engineering and Management ที่ Asian Institute of Technology (A.I.T) ในปี พ.ศ. 2531 และจบการศึกษาเมื่อ พ.ศ. 2532 เมื่อปี พ.ศ. 2529 ในขณะที่กำลังศึกษาอยู่ที่จุฬาลงกรณ์มหาวิทยาลัยนั้น ได้รับทุนผู้ช่วยสอนจากจุฬาลงกรณ์มหาวิทยาลัย ปัจจุบันดำรงตำแหน่งอาจารย์ประจำภาควิชาวิศวกรรมอุตสาหการ จุฬาลงกรณ์มหาวิทยาลัย