REFERENCES

1. Jaswon, M. A., and Maiti, M., " An Integral Equation Formulation of Plate Bending Problems," J. Engng. Math., Vol.2, 83-93, 1968.

2. Maiti, M., and Chakrabarty, S. K., " Integral Equation Solutions for Simply Supported Polygonal Plates," Int. J. Engng. Sci., Vol. 12, 793-806, 1974..

3. Altiero, N. J., and Sikarskie, D. L., " A Boundary Integral Method Applied to Plates of Arbitrary Plan Form," Computers and Structures, Vol. 9, 163-168. 1978.

4. Wu, B. C., and Altiero, N. J., "A Boundary Integral Method Applied to Plates of Arbitrary Plan Form and Arbitrary Boundary Conditions," Computers and Structures, Vol. 10, 703-707, 1979.

5. Stern, M., " A General Boundary Integral Formulation for the Numerical Solution of Plate Bending Problems," Int. J. Solids Structures, Vol. 15, 769-782, 1979.

6. Tottenham, H., "The Boundary Element Method for Plates and Shells," Developments in Boundary Element Methods (Banerjee, P. K., and Butterfield, R., eds.), Vol. 1, pp. 173-205, Applied Science Publishers, London, 1979.

7. Bezine, G., "A Boundary Integral Equation Method for Plate Flexure with Conditions inside the Domain," Int. J. Numer. Methods. Eng., Vol. 17, 1647-1657, 1981.

8. Kumar, K. S., "Boundary Integral Equations for Plate Bending," M.Eng. Thesis, Division of Structural Engineering and Construction, Asian Institute of Technology, 1982.

9. Banerjee, P. K., and Butterfield, R., Boundary Element Methods in Engineering Science, McGraw-Hill, England, 1st ed., 1981.

10. Timoshenko, S. P., and Woinowsky-Krieger, S., Theory of Plates and Shells, McGraw-Hill, New York, 2nd ed., 1959.

11. Oden, J. T.,Mechanics of Elastic Structures, McGraw-Hill, New York, 1st ed., 1967.

FIGURE

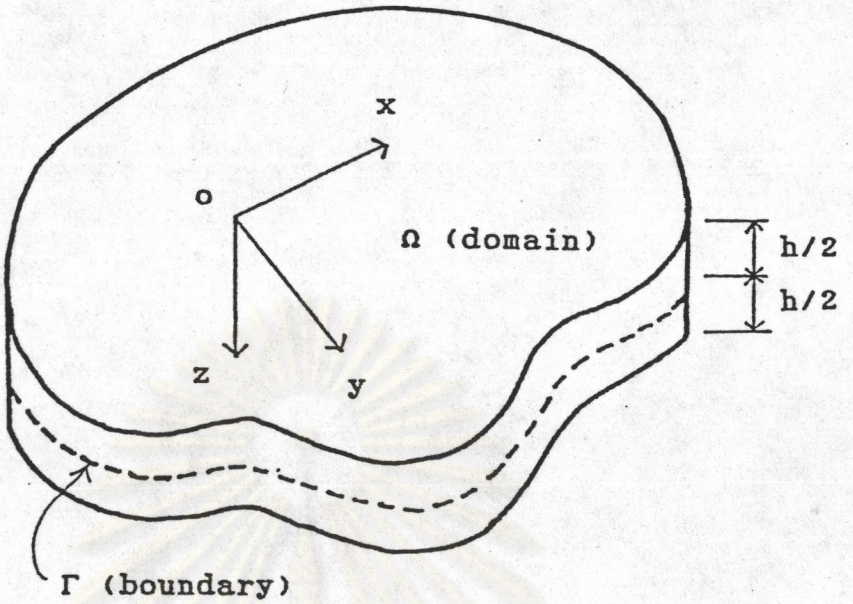


FIGURE 1 Element of plate

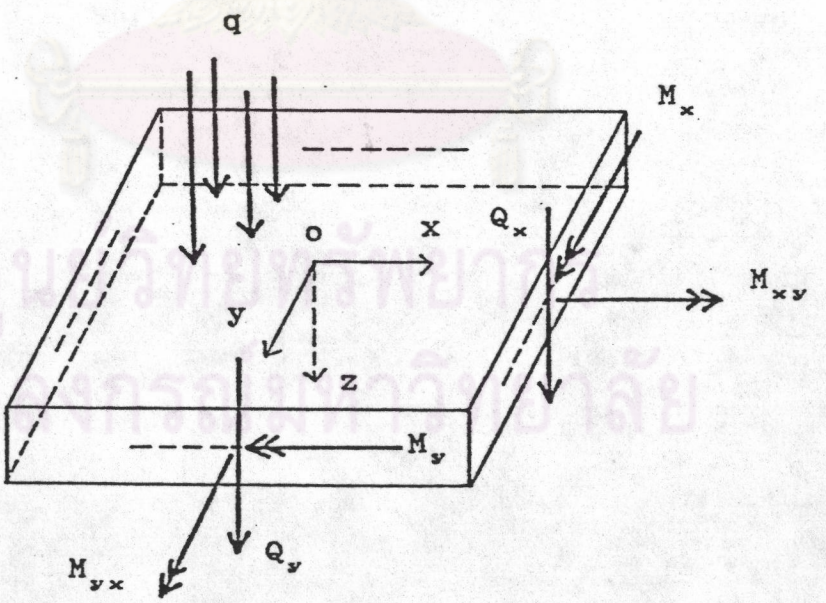


FIGURE 2 Sign convention of stress resultants

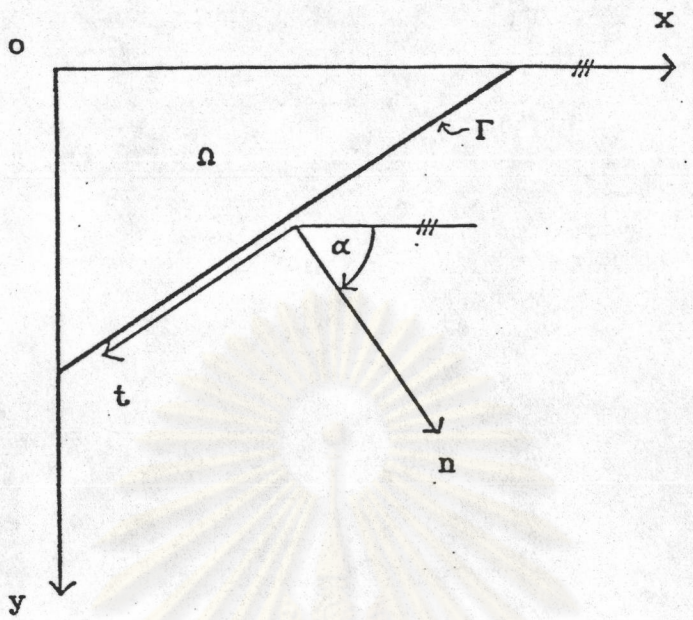


FIGURE 3 Normal co-ordinates

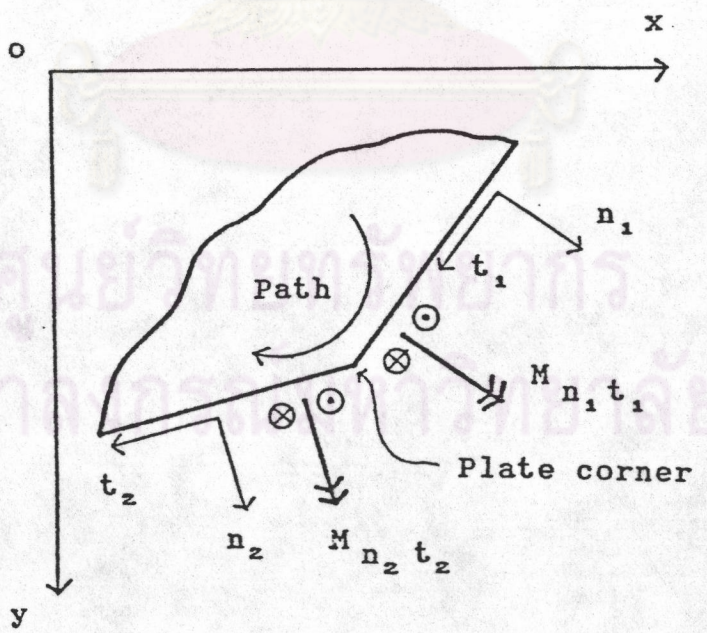
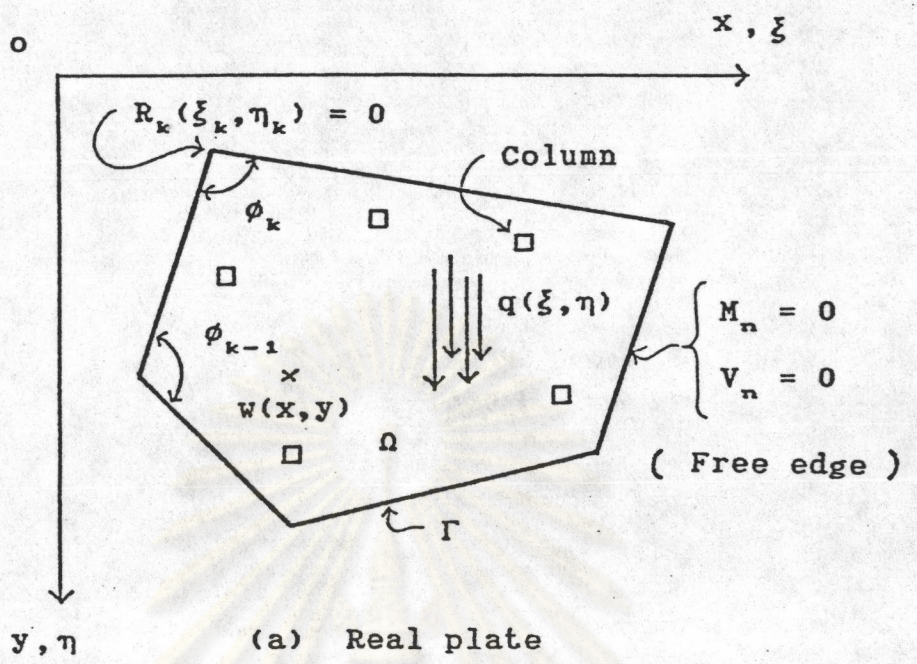
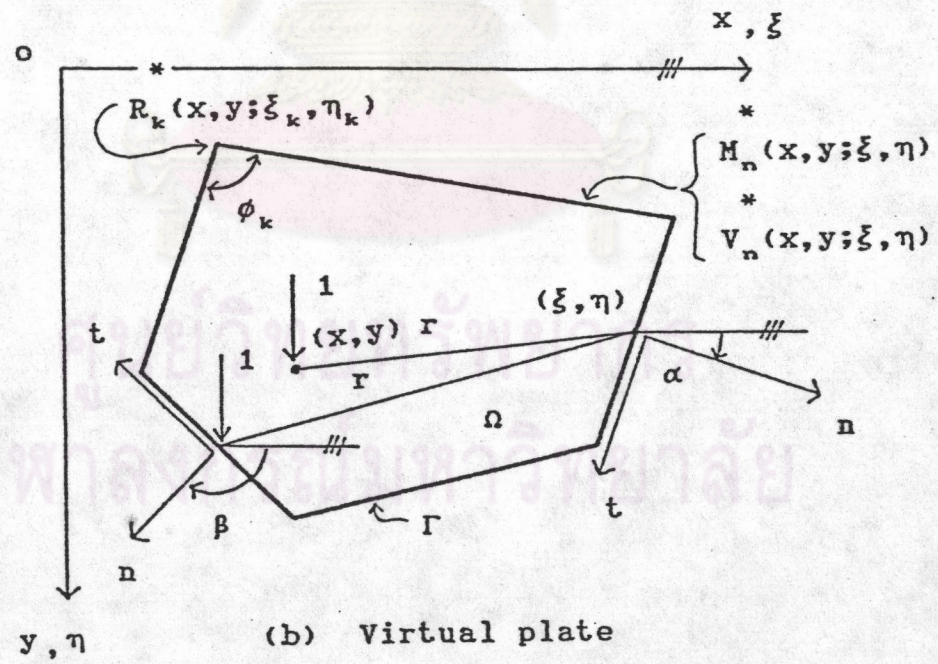


FIGURE 4 Representation of corner force



(a) Real plate



(b) Virtual plate

FIGURE 5 Force and displacement systems in Betti's theorem

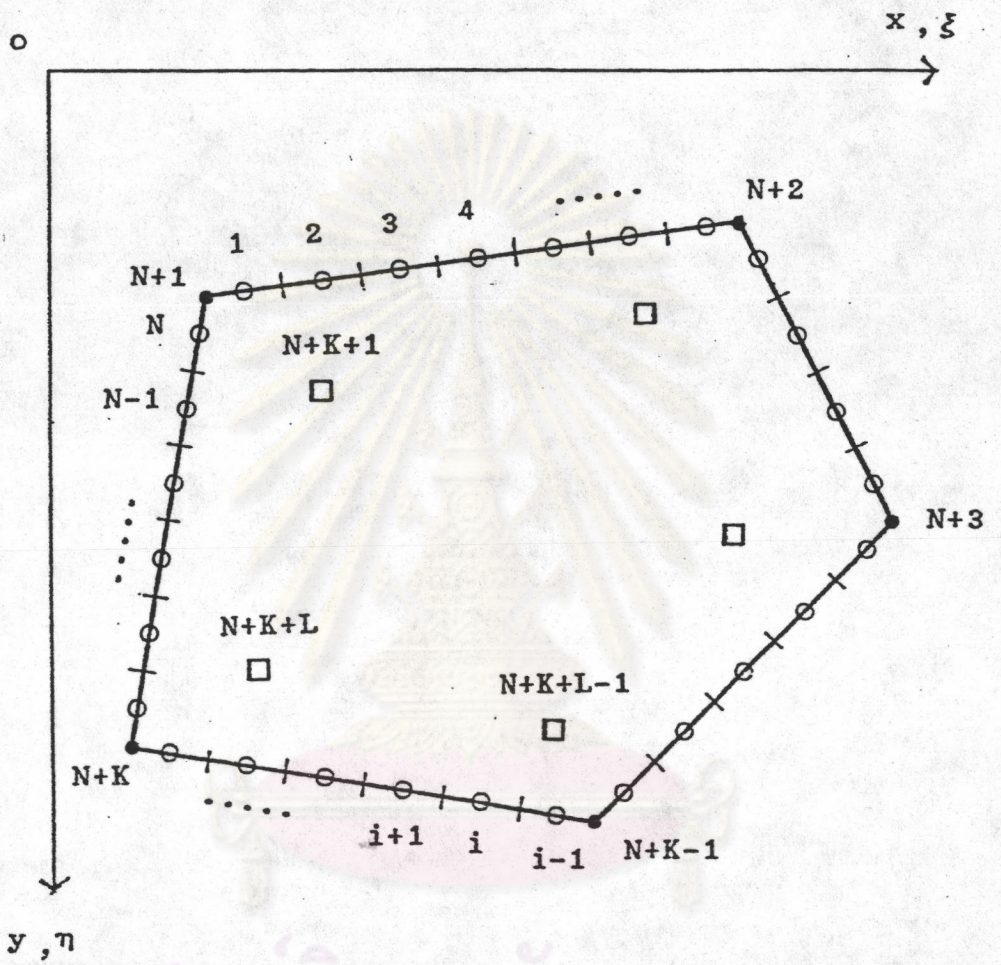


FIGURE 6 Subdivision of the boundary

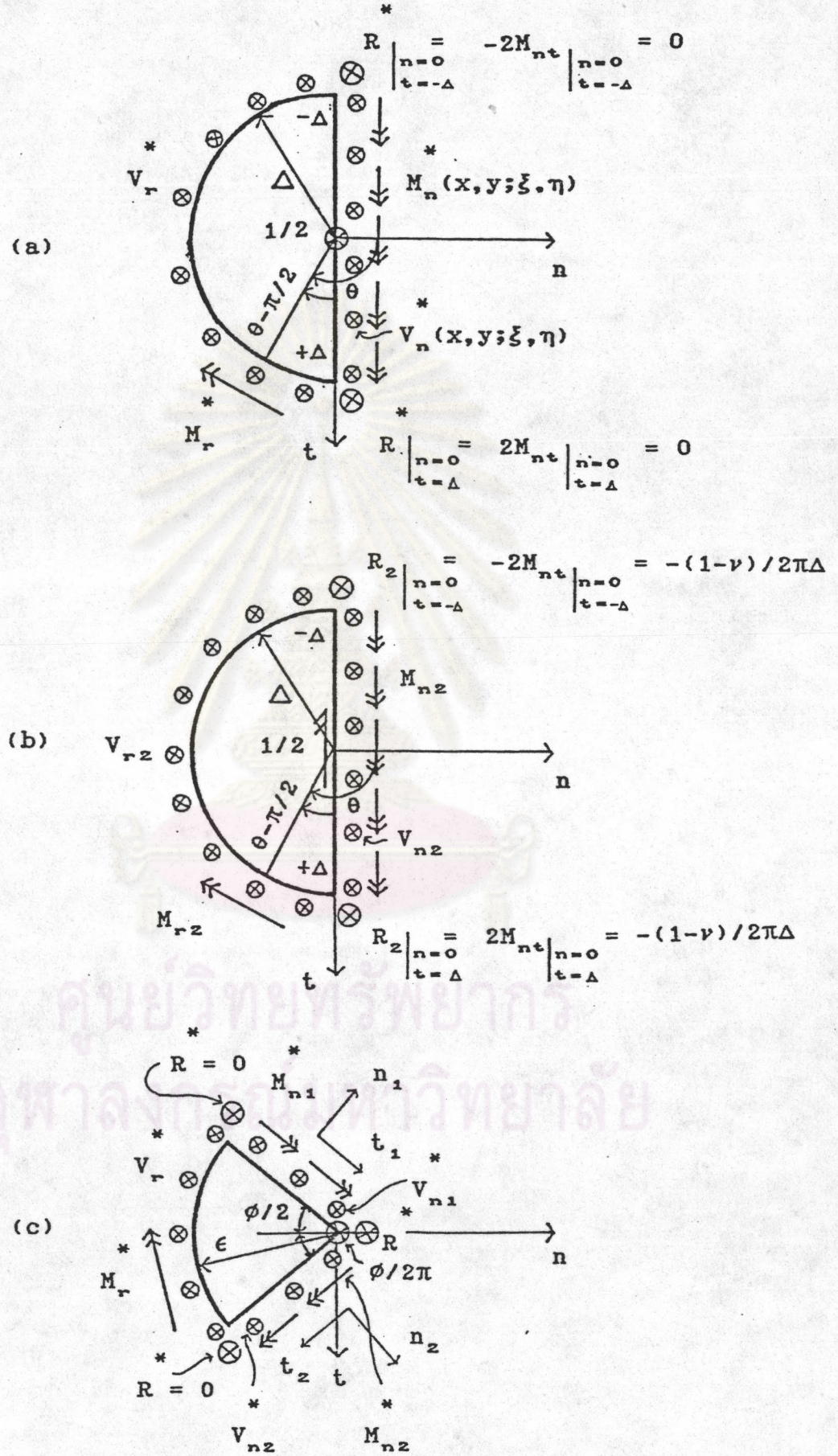


FIGURE 7 Force system of the free-body circular sector element

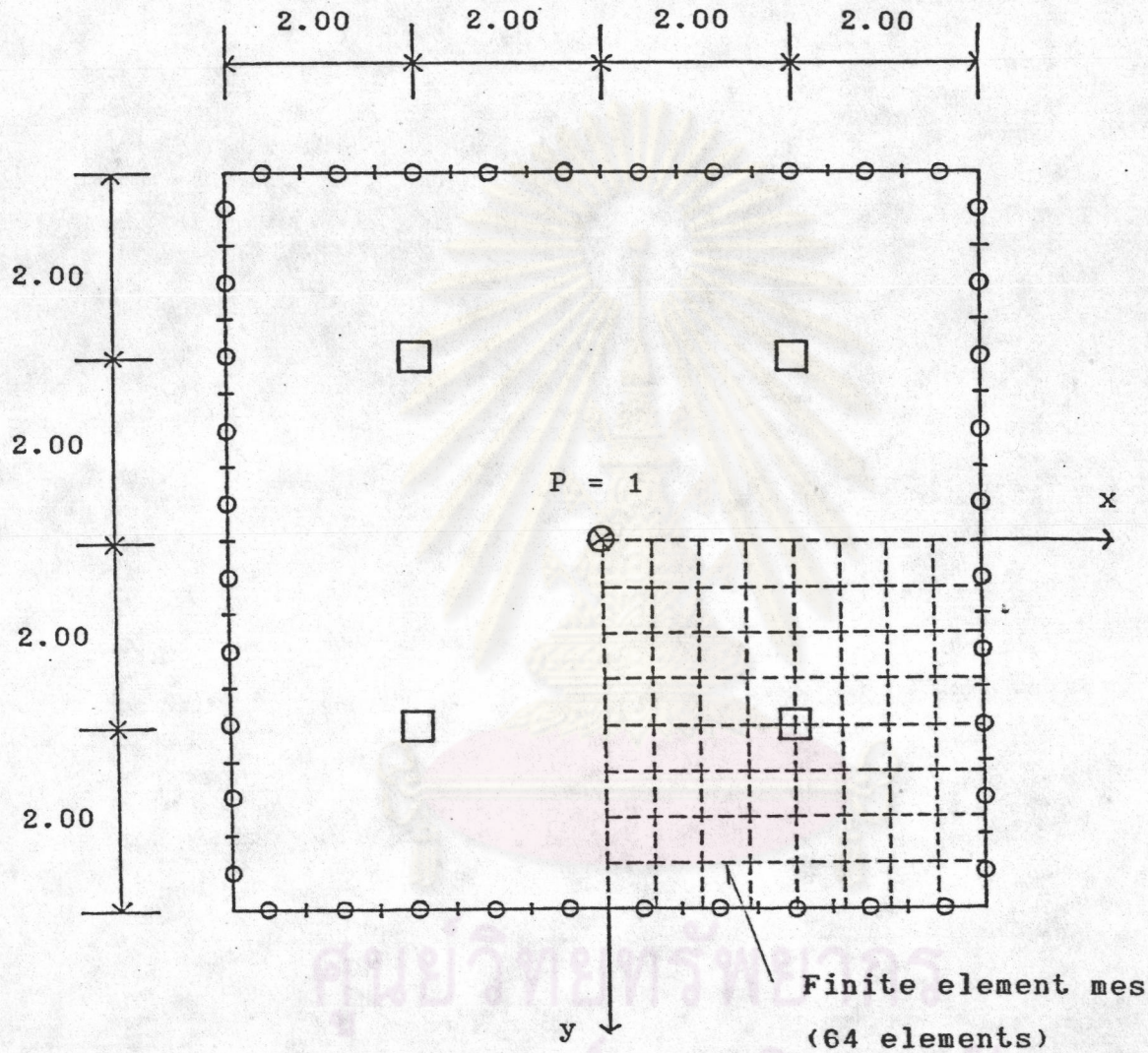


Plate rigidity , D = 1.00

Poisson's ratio , ν = 0.30

Number of boundary subintervals = 10 per side of plate

FIGURE 8 Square plate with four rigid supports , Example 1 .

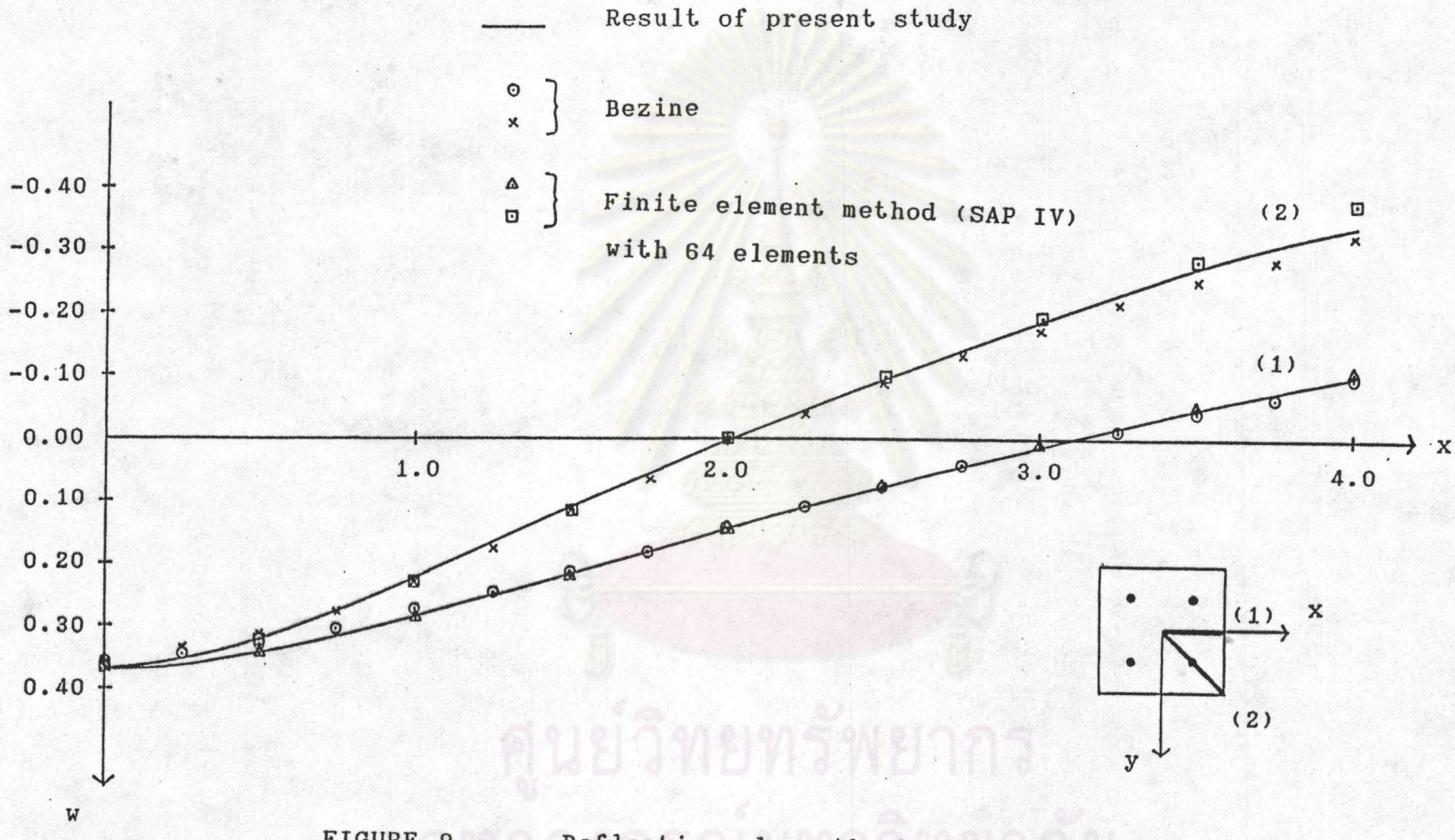


FIGURE 9 Deflection along the horizontal line of symmetry and the diagonal

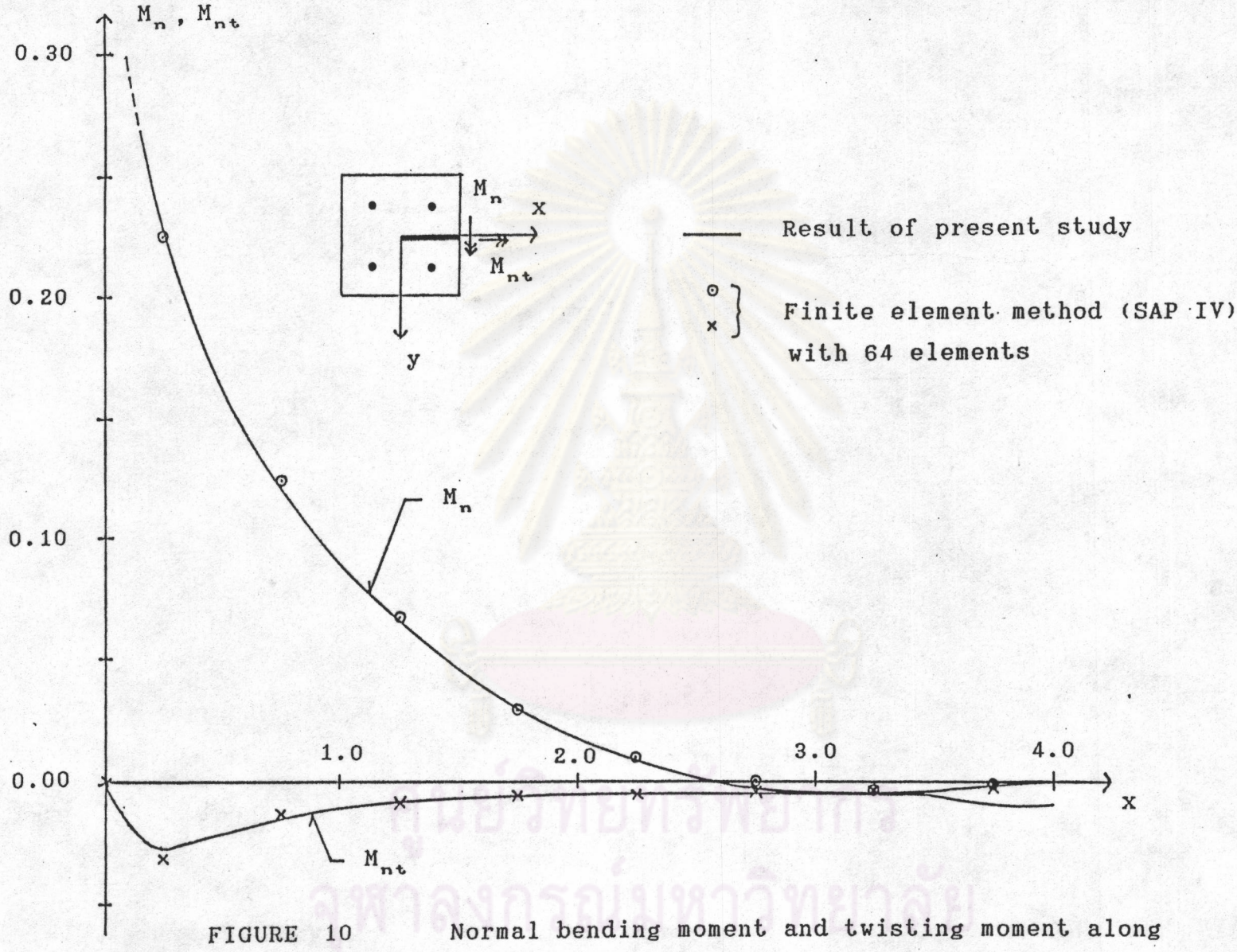


FIGURE 10 Normal bending moment and twisting moment along the horizontal line (y = 0.25)

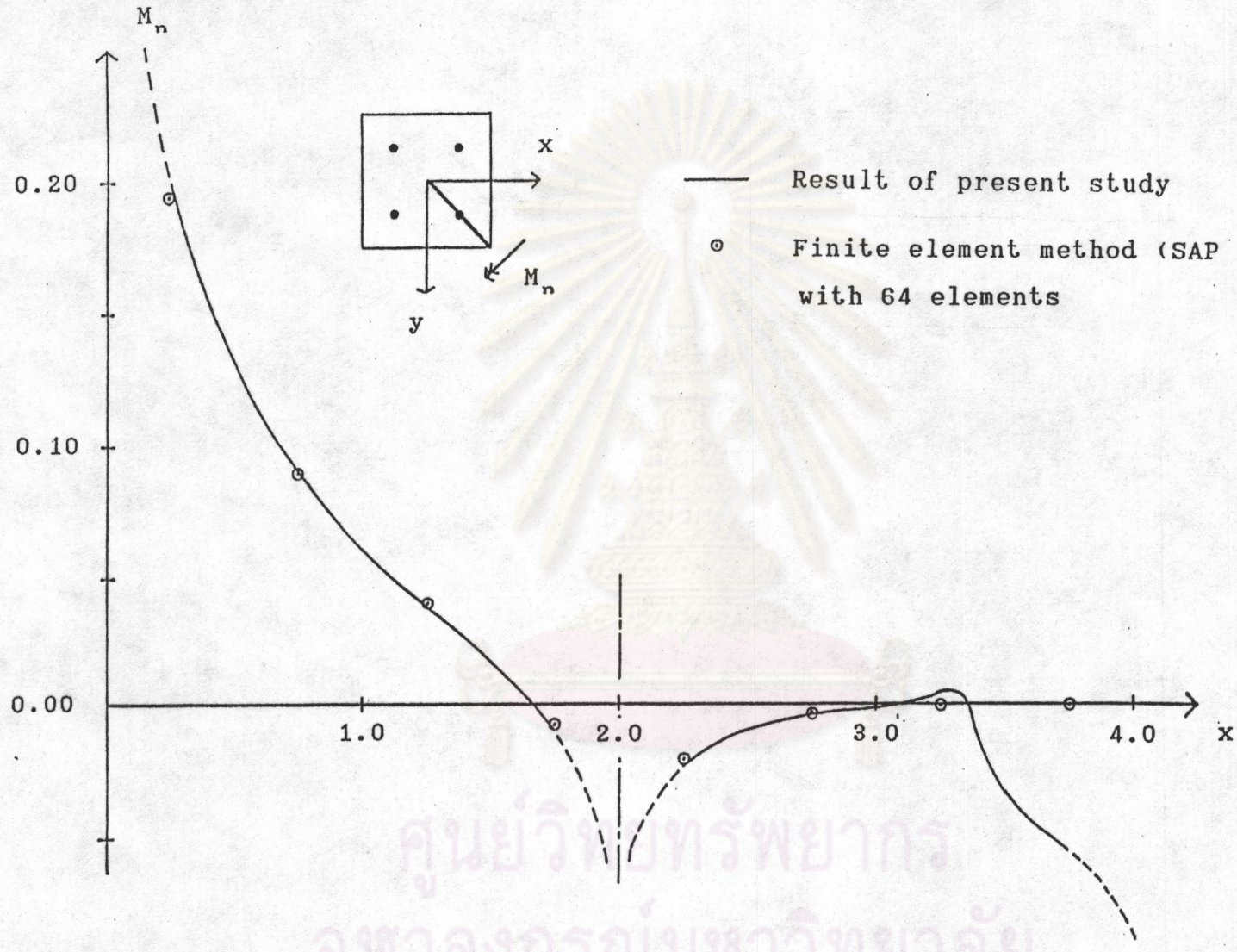
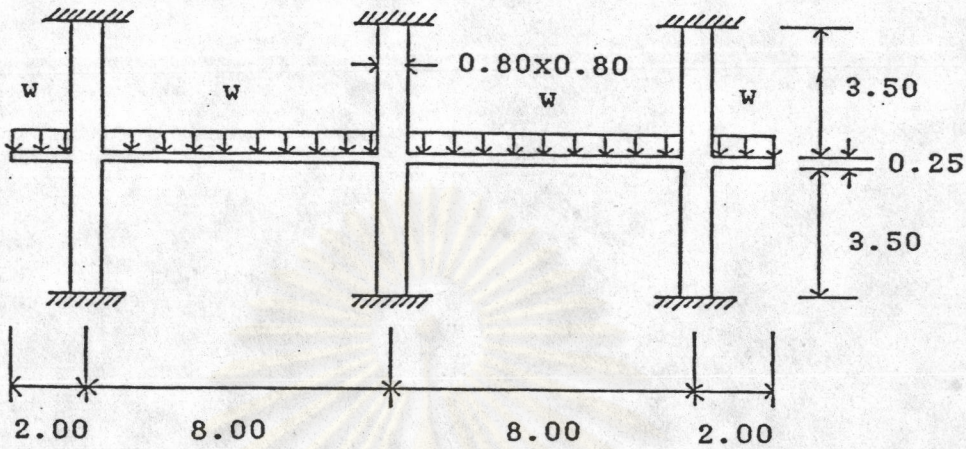


FIGURE 11 Normal bending moment along the diagonal

$E = 2.204 \times 10^{9}$ kg/m^2

$\nu = 0.15$

$w = 1520$ kg/m^2



Number of boundary subintervals = 10 per side of plate

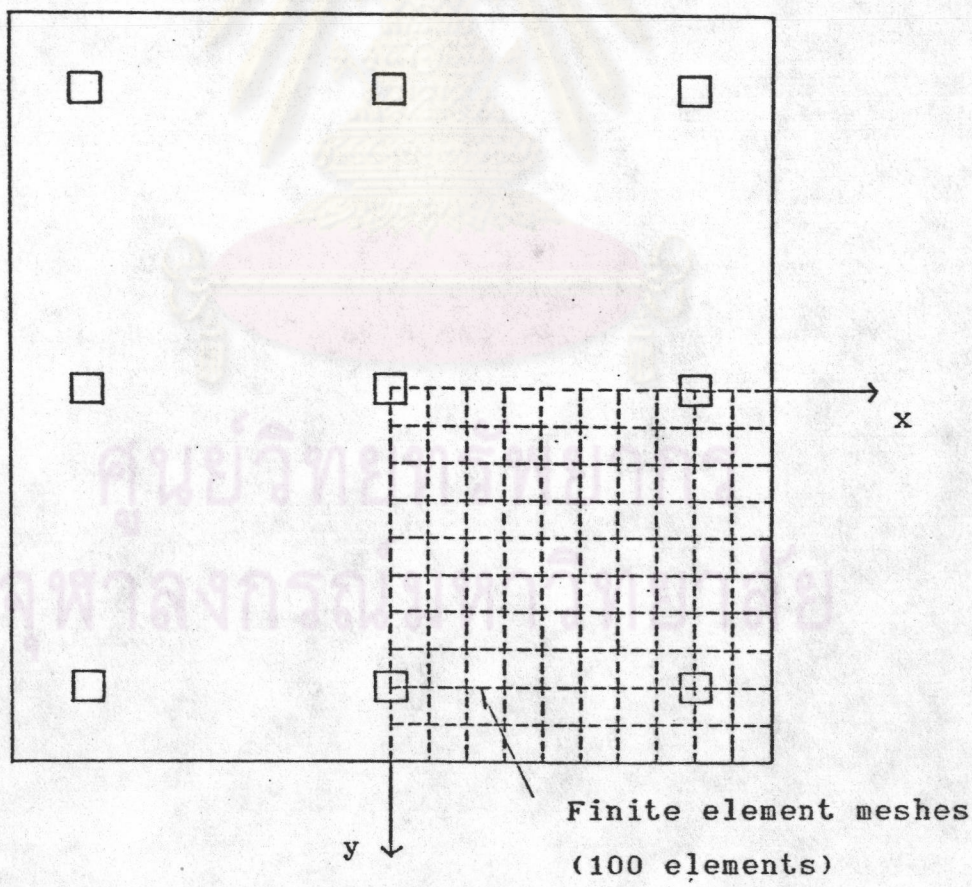


FIGURE 12 Flat plate , Example 2 .

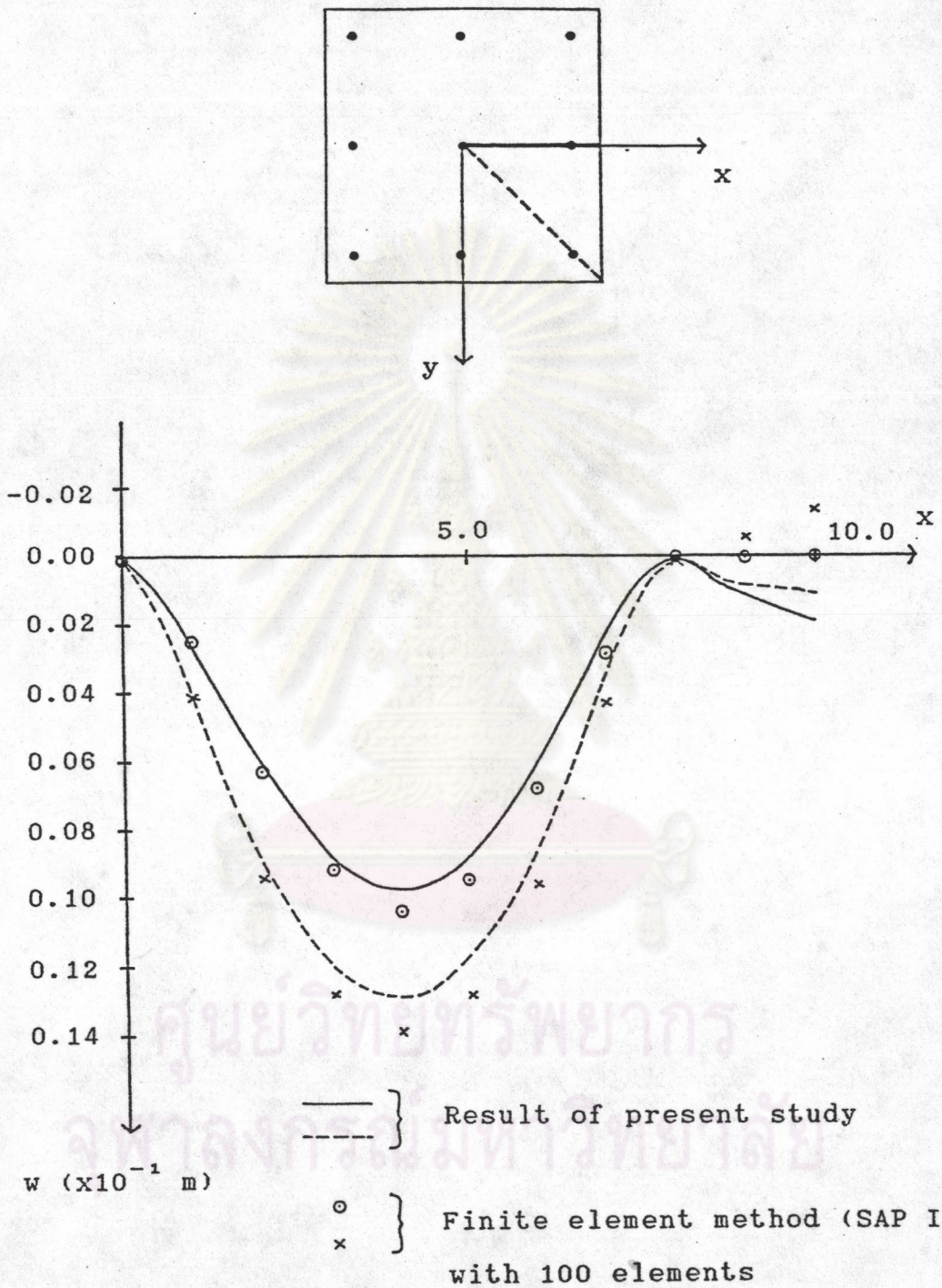


FIGURE 13 Deflection along the horizontal line of symmetry and the diagonal

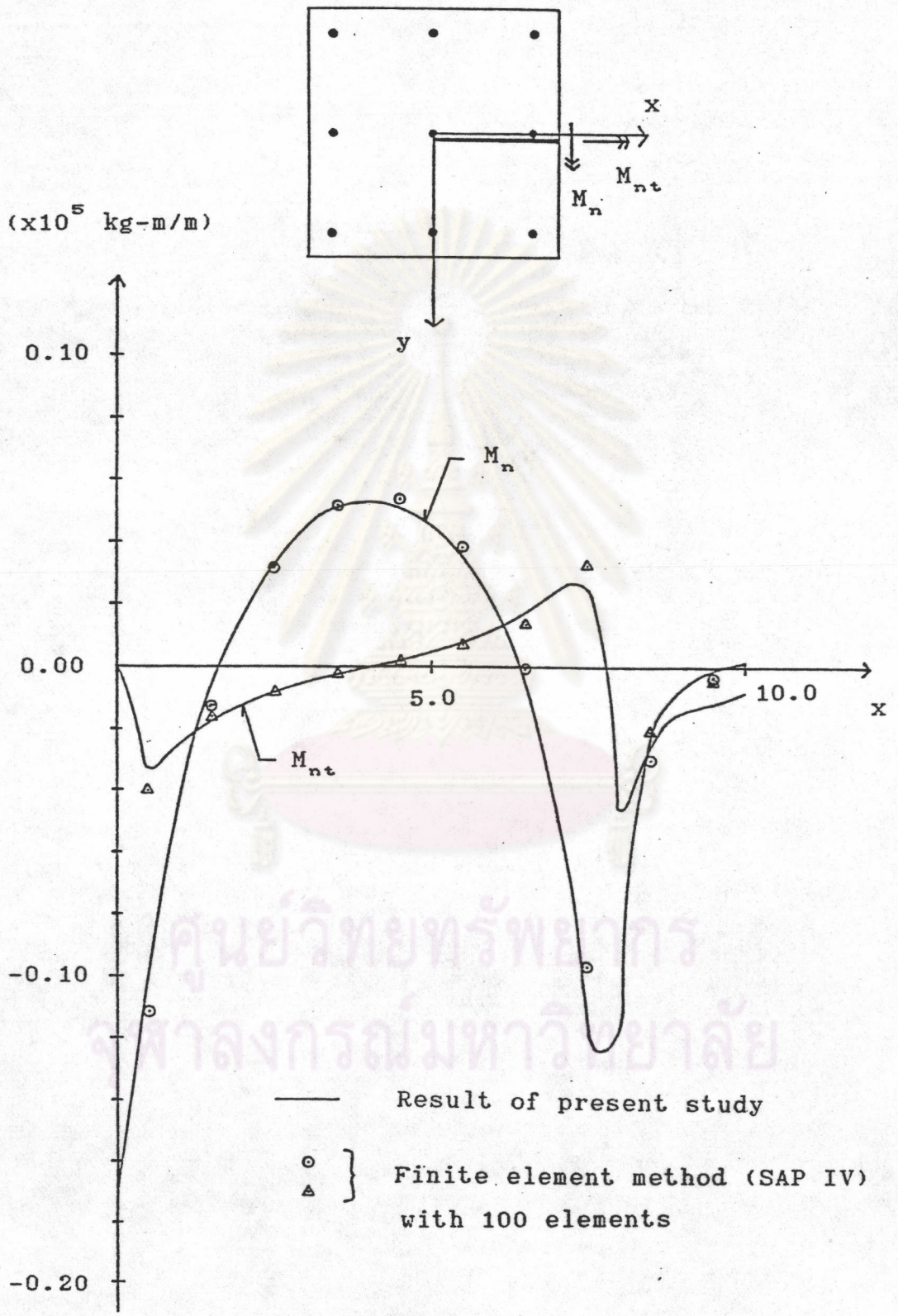


FIGURE 14 Normal bending moment and twisting moment along the horizontal line (y = 0.5)

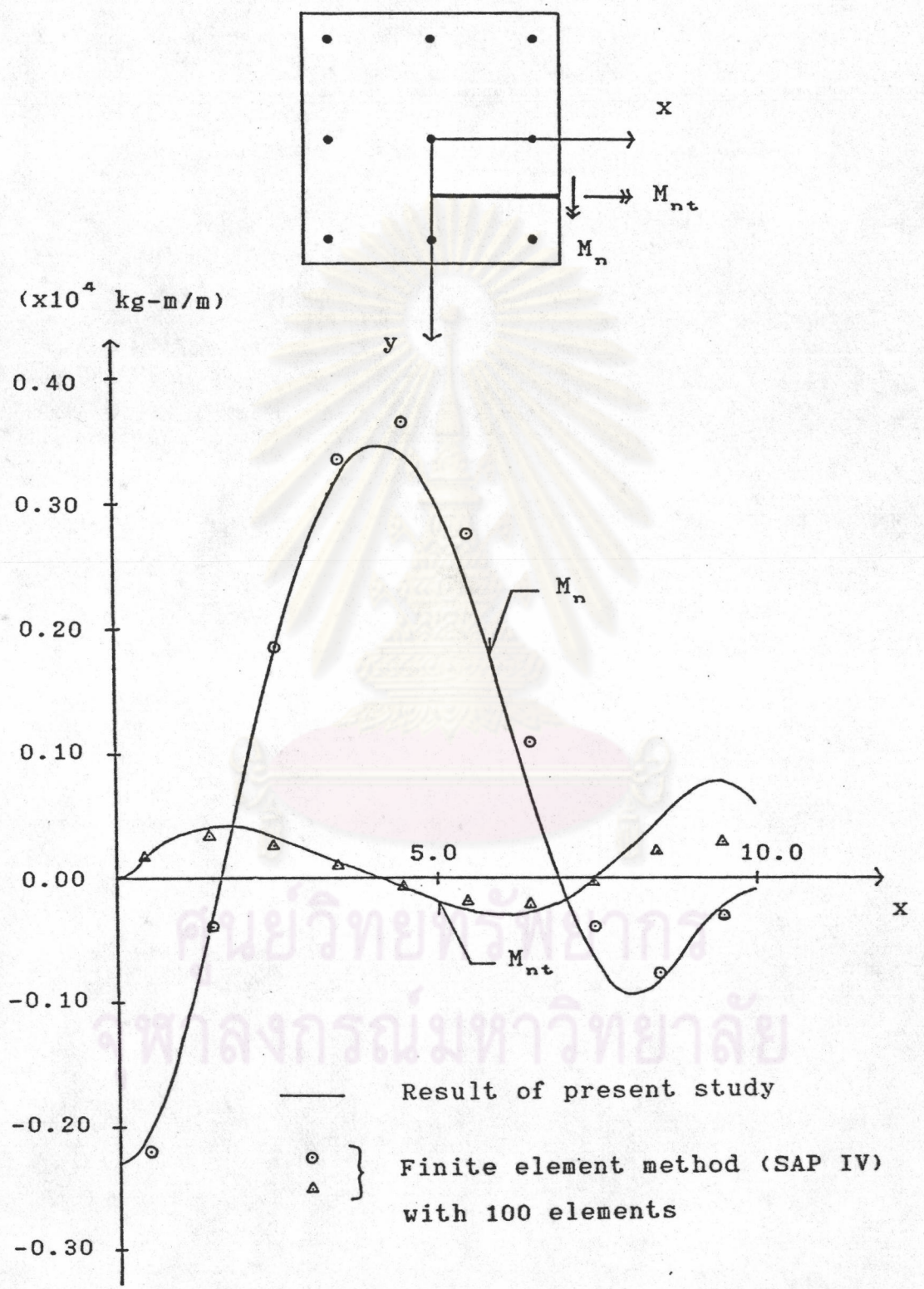


FIGURE 15 Normal bending moment and twisting moment along the horizontal line (y = 4.5)

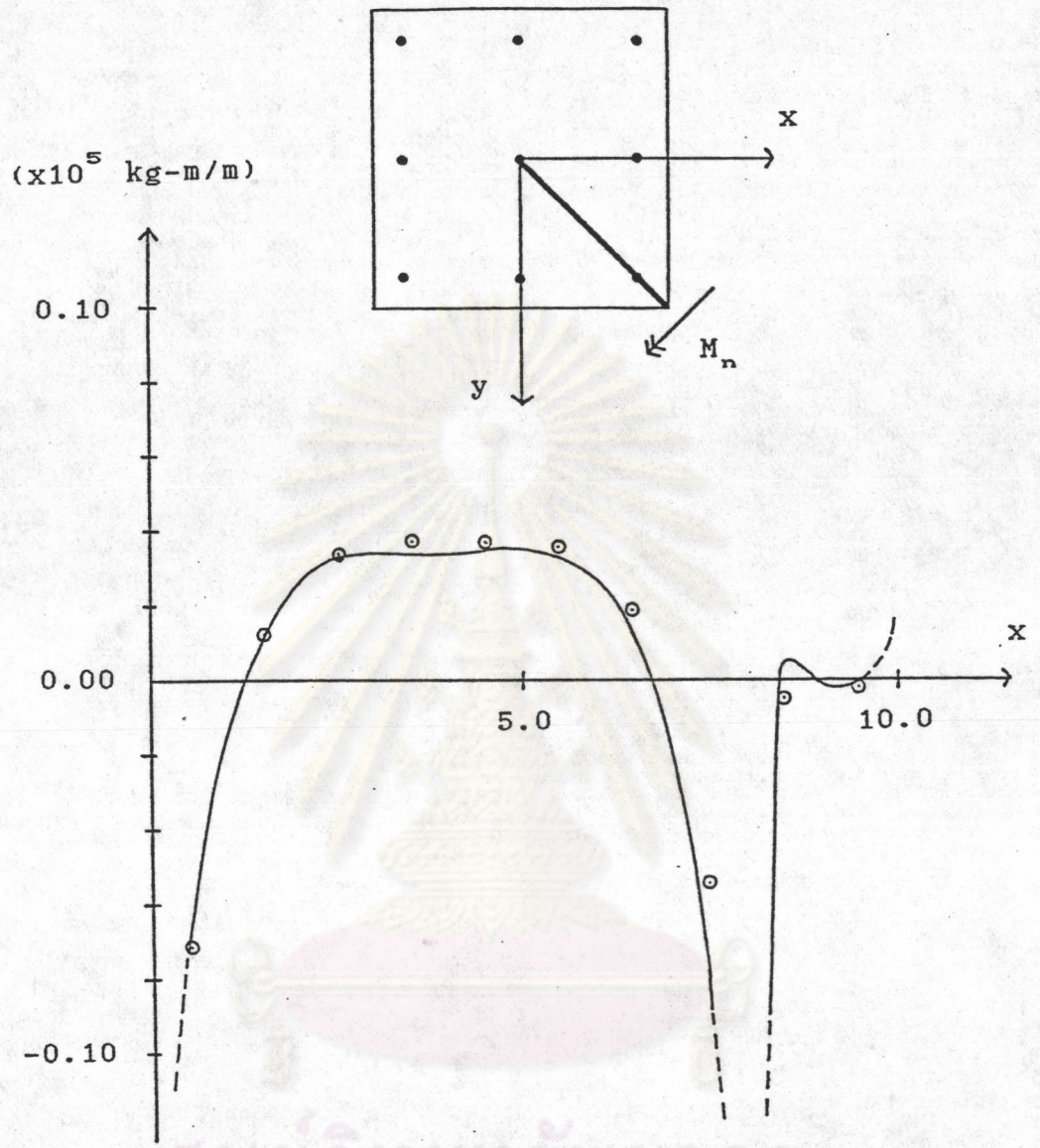
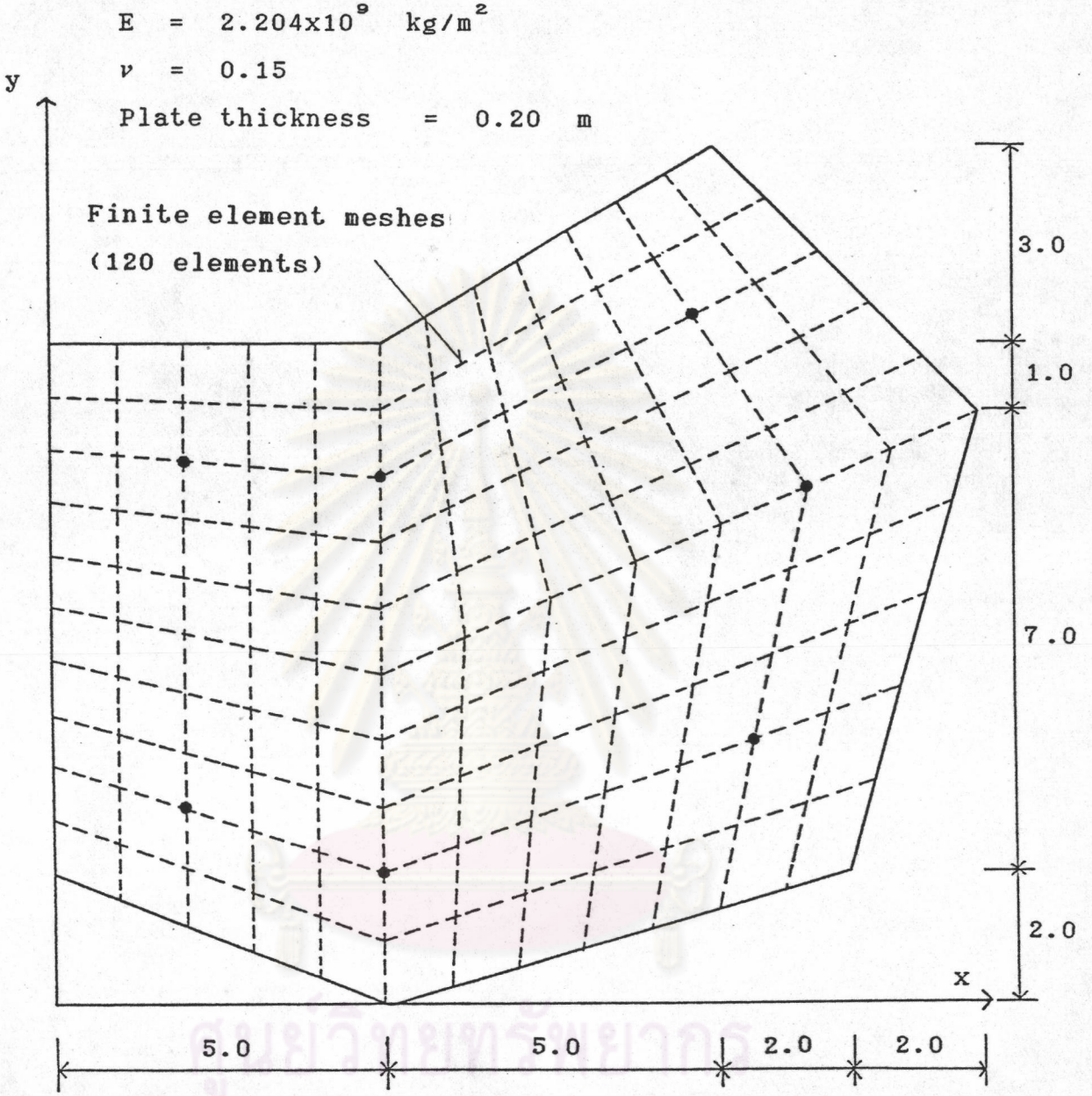


FIGURE 16 Normal bending moment along the diagonal



Column size = 0.40x0.40 m

Column height = 3.50 m

Uniformly distributed load = 680 kg/m²

Number of boundary subintervals = 41

FIGURE 17 Rectilinear plate , Example 3 .

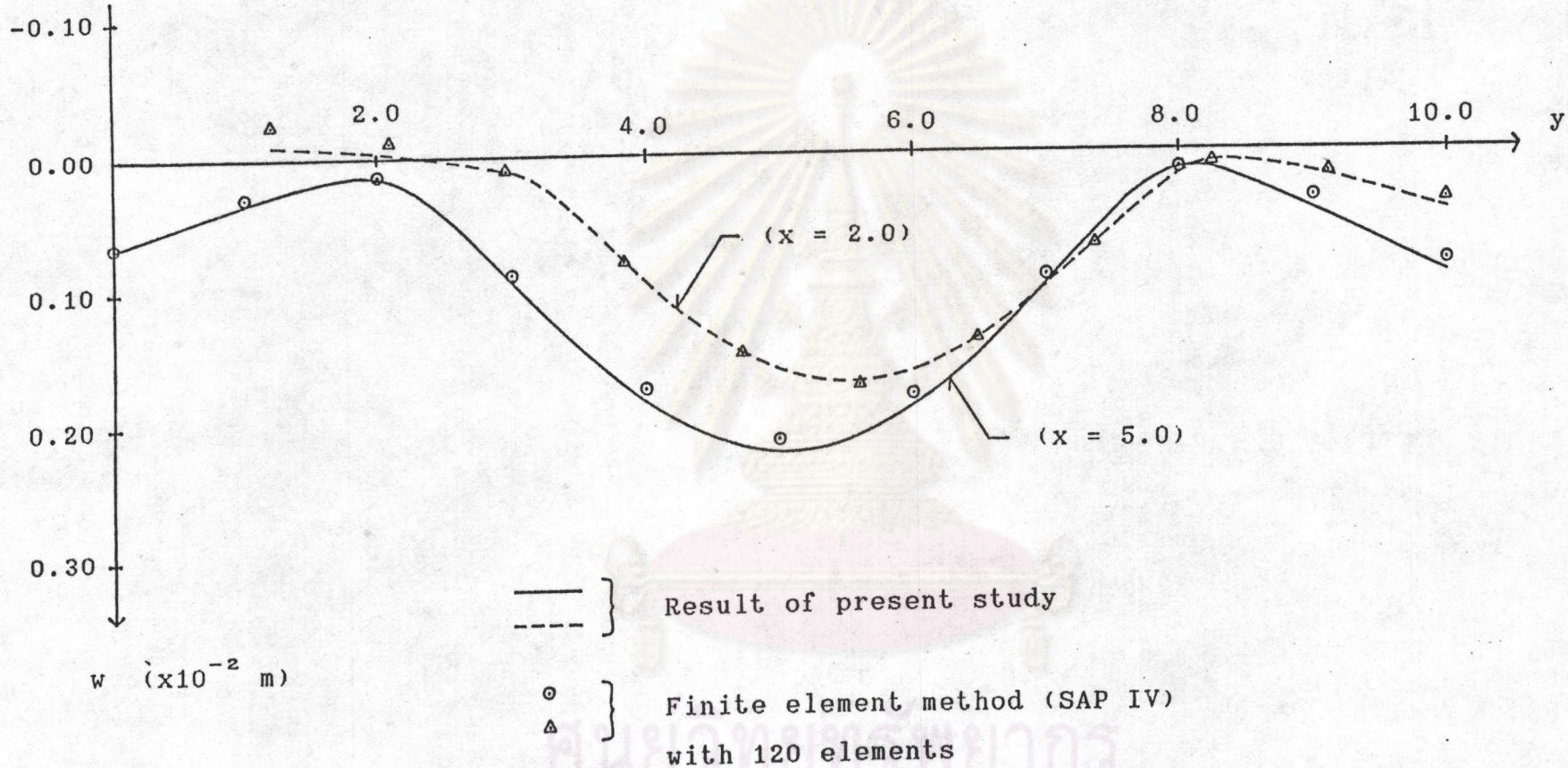


FIGURE 18 Deflection along the vertical line
x = 2.0 and x = 5.0

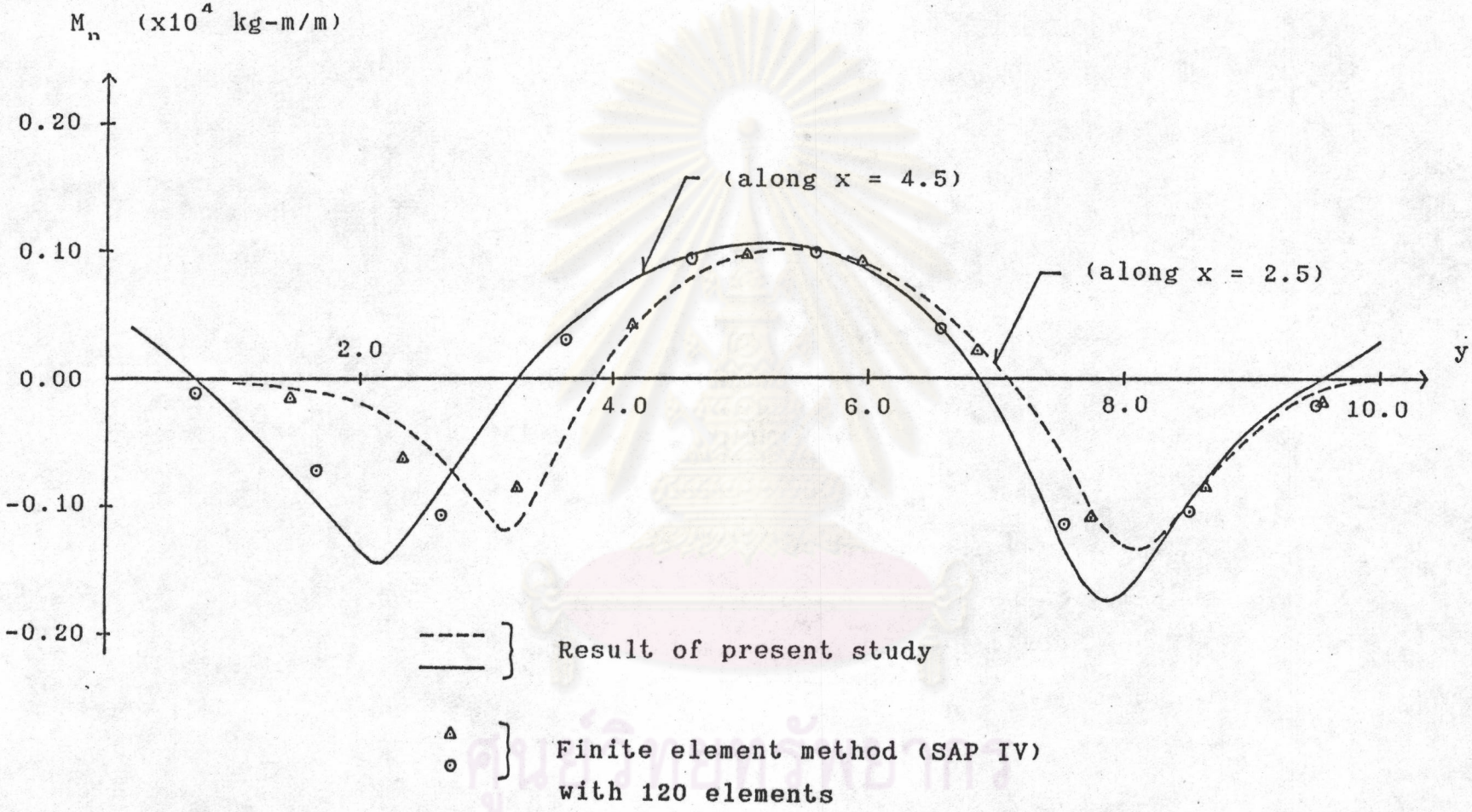


FIGURE 19 Normal bending moment along the vertical line x = 2.5 and x = 4.5

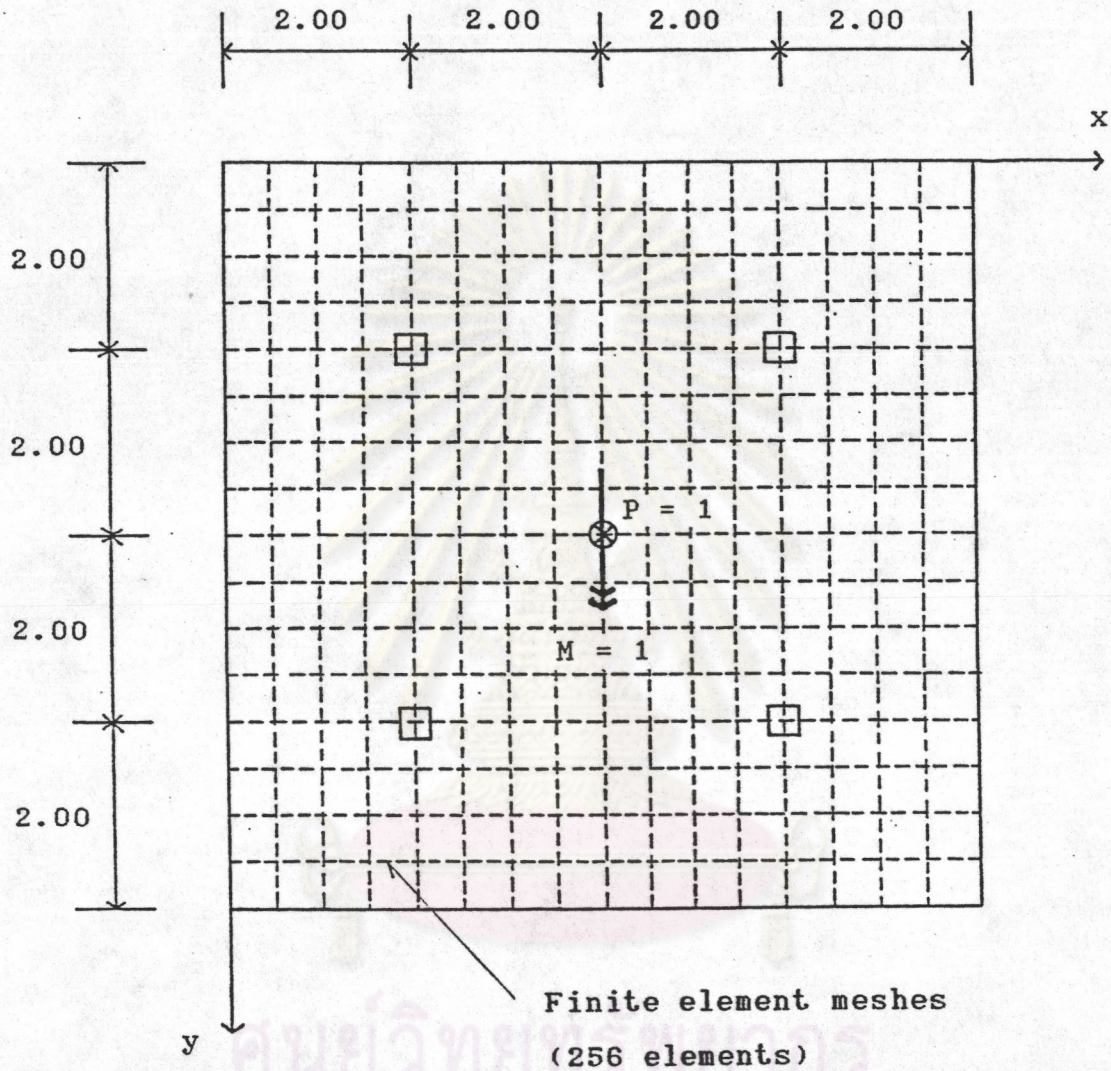


Plate rigidity , D = 1.00

Poisson's ratio , ν = 0.30

Number of boundary subintervals = 10 per side of plate

FIGURE 20 Square plate with four rigid supports, Example 4 .

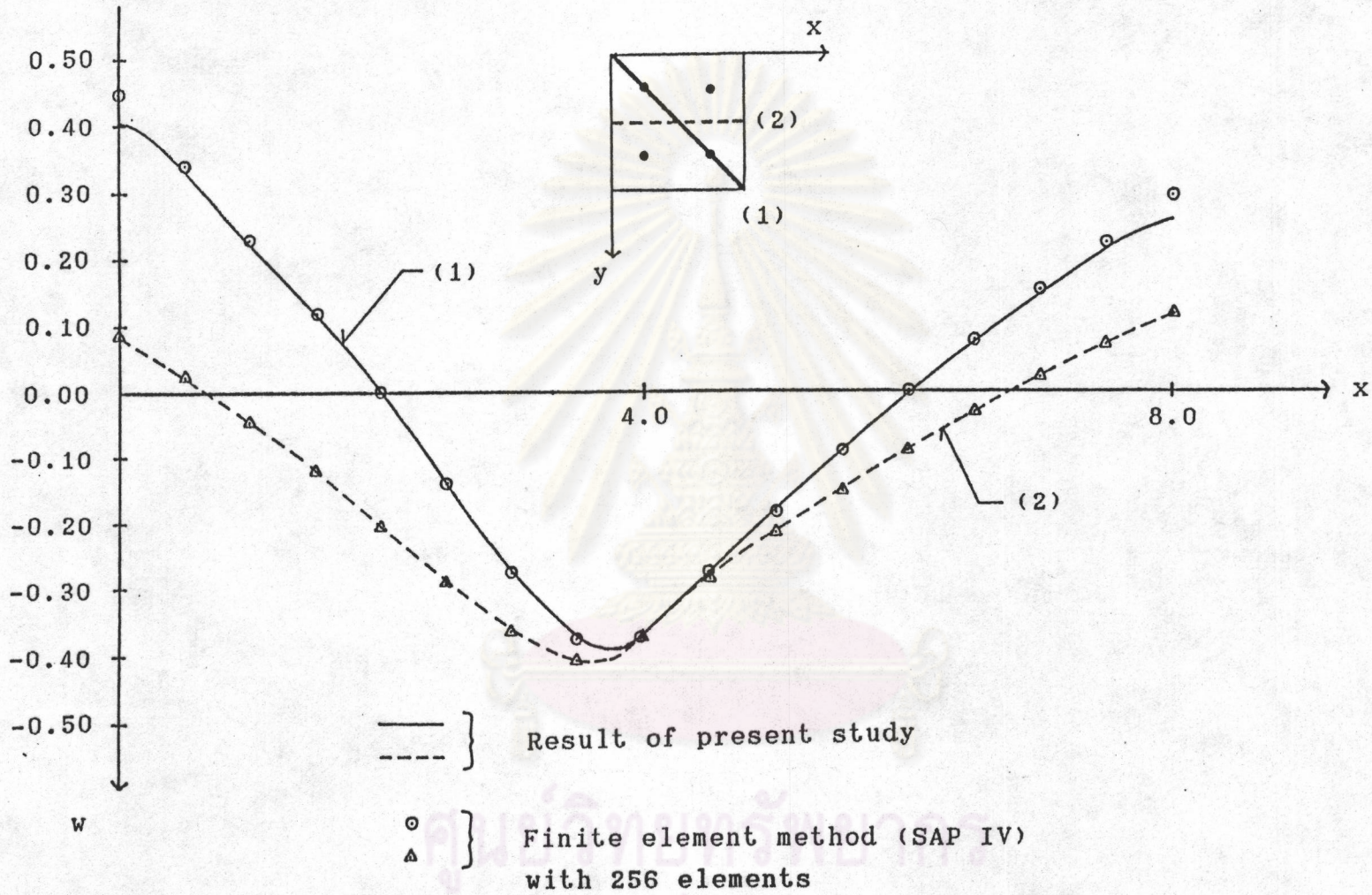


FIGURE 21 Deflection along the horizontal line and the diagonal

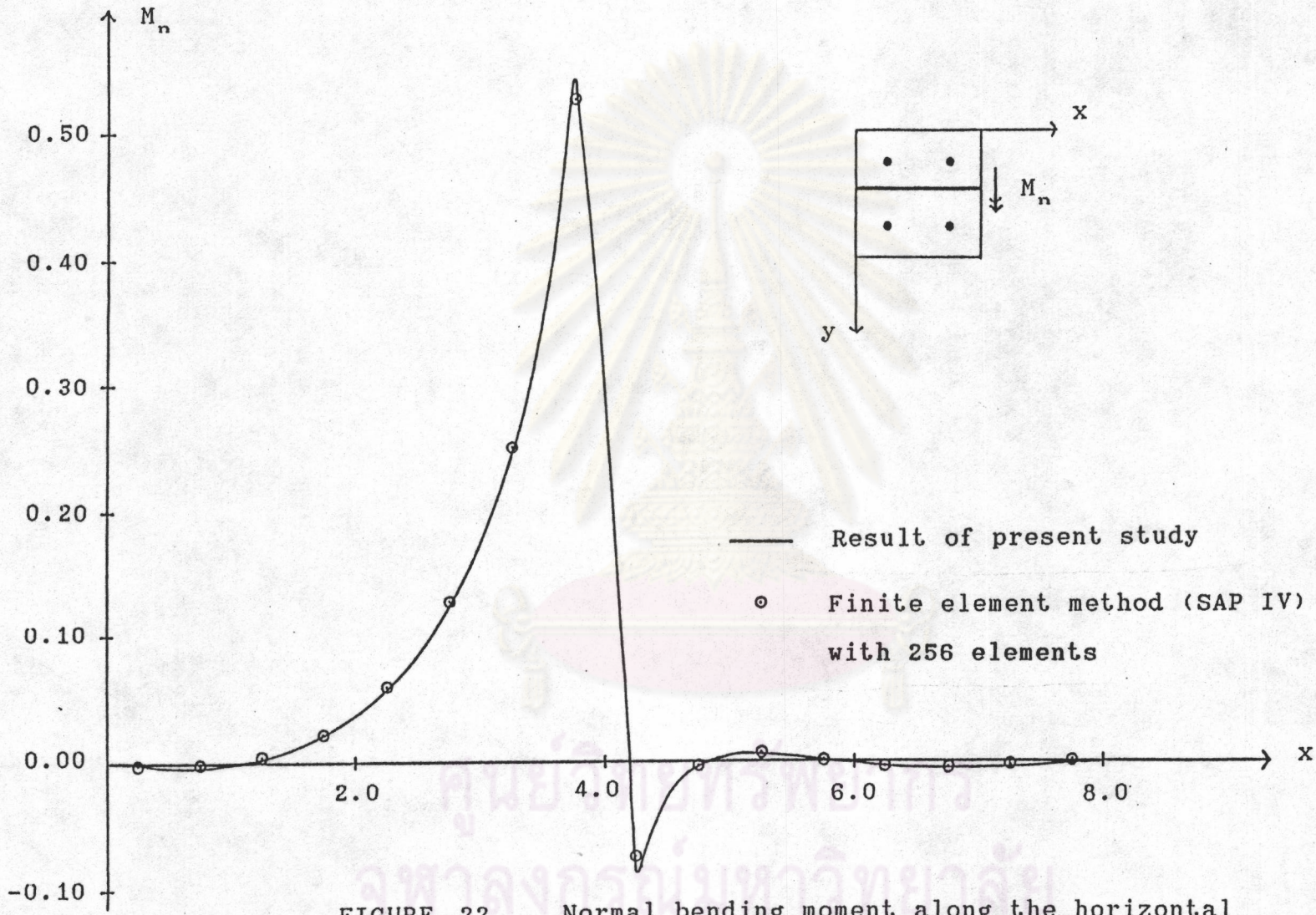


FIGURE 22 Normal bending moment along the horizontal line (y = 3.75)

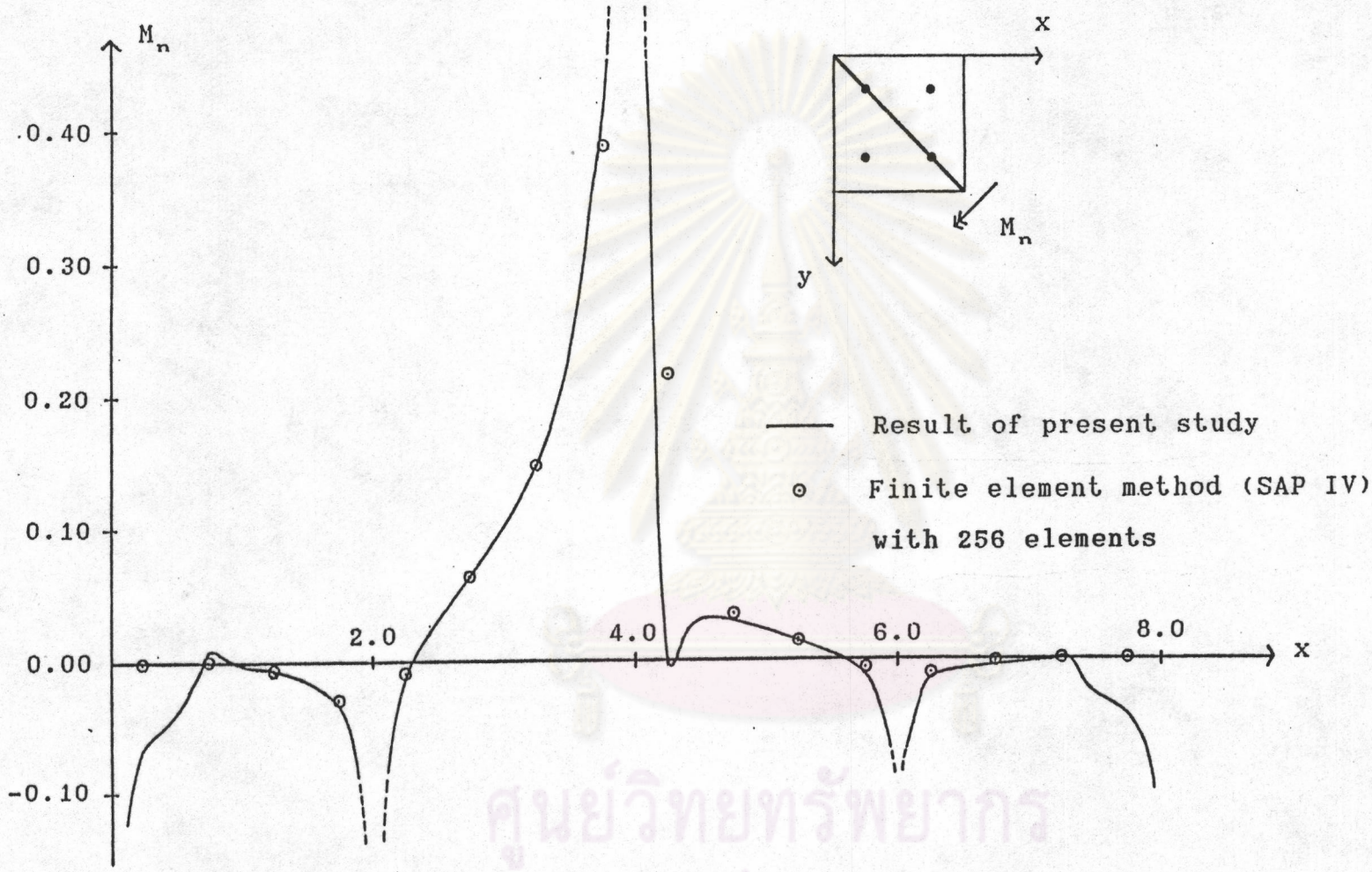


FIGURE 23 Normal bending moment along the diagonal

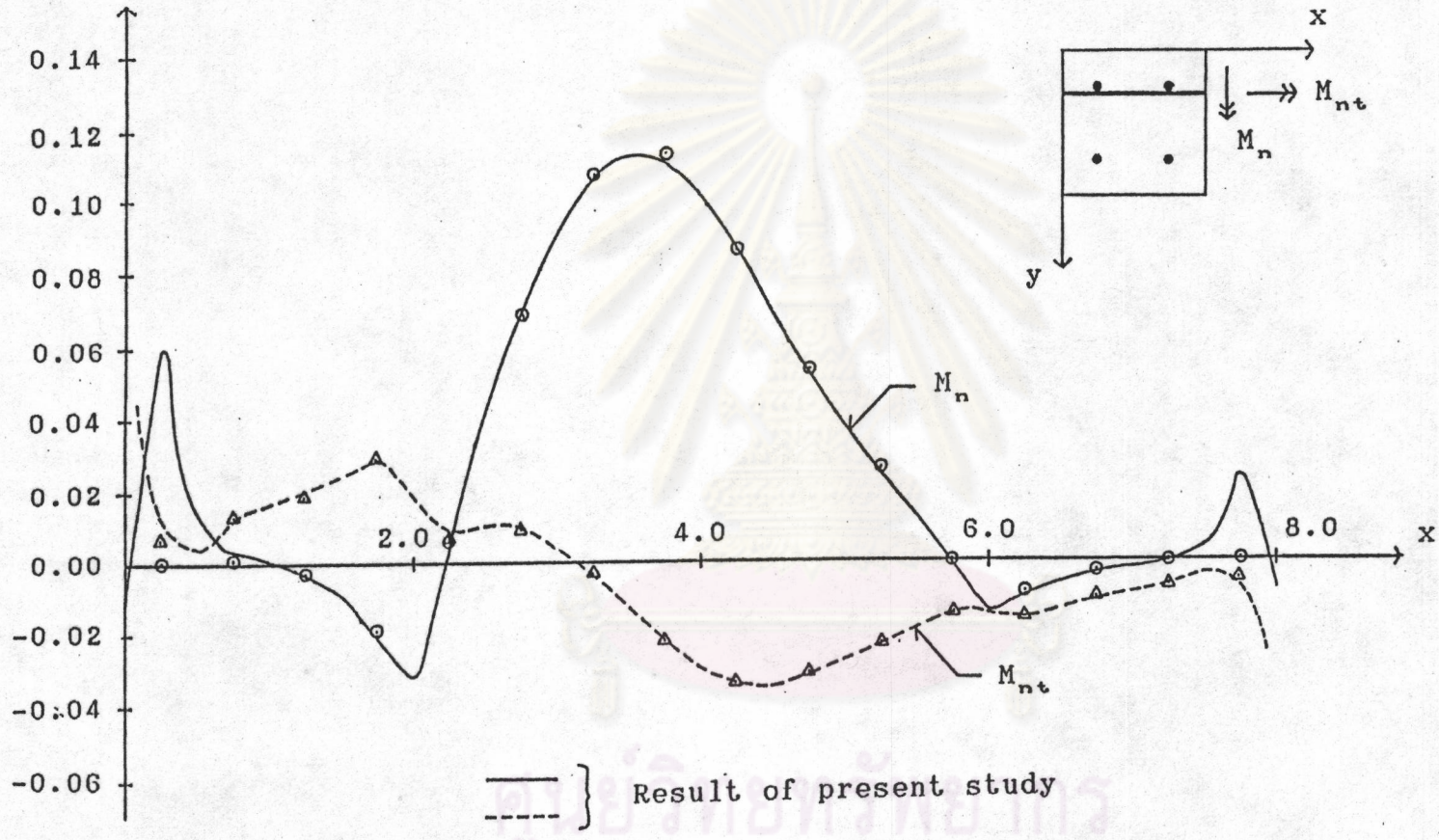


FIGURE 24 Normal bending moment and twisting moment along the horizontal line (y = 2.25)

APPENDIX

APPENDIX A

Transformation of Co-ordinates (Fig.3)

First order derivatives :

$$\begin{Bmatrix} \frac{\partial w}{\partial n} \\ \\ \frac{\partial w}{\partial t} \end{Bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha \\ \\ -\sin\alpha & \cos\alpha \end{bmatrix} \begin{Bmatrix} \frac{\partial w}{\partial x} \\ \\ \frac{\partial w}{\partial y} \end{Bmatrix}$$

Second order derivatives :

$$\begin{Bmatrix} \frac{\partial^2 w}{\partial n^2} \\ \\ \frac{\partial^2 w}{\partial n \partial t} \\ \\ \frac{\partial^2 w}{\partial t^2} \end{Bmatrix} = \begin{bmatrix} \cos^2\alpha & \sin 2\alpha & \sin^2\alpha \\ \\ \frac{-\sin 2\alpha}{2} & \cos 2\alpha & \frac{\sin 2\alpha}{2} \\ \\ \sin^2\alpha & -\sin 2\alpha & \cos^2\alpha \end{bmatrix} \begin{Bmatrix} \frac{\partial^2 w}{\partial x^2} \\ \\ \frac{\partial^2 w}{\partial x \partial y} \\ \\ \frac{\partial^2 w}{\partial y^2} \end{Bmatrix}$$

Third order derivatives :

$$\begin{Bmatrix} \frac{\partial^3 w}{\partial n^3} \\ \frac{\partial^3 w}{\partial n^2 \partial t} \\ \frac{\partial^3 w}{\partial n \partial t^2} \\ \frac{\partial^3 w}{\partial t^3} \end{Bmatrix} =$$

$$\begin{bmatrix} \cos^3\alpha & \frac{3\cos\alpha\sin 2\alpha}{2} & \frac{3\sin\alpha\sin 2\alpha}{2} & \sin^3\alpha \\ \frac{-\cos\alpha\sin 2\alpha}{2} & \cos^3\alpha-\sin\alpha\sin 2\alpha & -\sin^3\alpha+\cos\alpha\sin 2\alpha & \frac{\sin\alpha\sin 2\alpha}{2} \\ \frac{\sin\alpha\sin 2\alpha}{2} & \sin^3\alpha-\cos\alpha\sin 2\alpha & \cos^3\alpha-\sin\alpha\sin 2\alpha & \frac{\cos\alpha\sin 2\alpha}{2} \\ -\sin^3\alpha & \frac{3\sin\alpha\sin 2\alpha}{2} & \frac{-3\cos\alpha\sin 2\alpha}{2} & \cos^3\alpha \end{bmatrix} \begin{Bmatrix} \frac{\partial^3 w}{\partial x^3} \\ \frac{\partial^3 w}{\partial x^2 \partial y} \\ \frac{\partial^3 w}{\partial x \partial y^2} \\ \frac{\partial^3 w}{\partial y^3} \end{Bmatrix}$$

APPENDIX B

Influence Functions (Fig.7)

$$w^*(x,y;\xi,\eta) = \frac{r^2 \ln r}{8\pi D}$$

$$\frac{\partial w^*(x,y;\xi,\eta)}{\partial x} = \frac{-1}{8\pi D}(\xi - x)(1 + 2\ln r)$$

$$\frac{\partial w^*(x,y;\xi,\eta)}{\partial y} = \frac{-1}{8\pi D}(\eta - y)(1 + 2\ln r)$$

$$\frac{\partial w^*(x,y;\xi,\eta)}{\partial \xi} = \frac{1}{8\pi D}(\xi - x)(1 + 2\ln r)$$

$$\frac{\partial w^*(x,y;\xi,\eta)}{\partial \eta} = \frac{1}{8\pi D}(\eta - y)(1 + 2\ln r)$$

$$\frac{\partial w^*(x,y;\xi,\eta)}{\partial n(x,y)} = \frac{-1}{8\pi D}(1 + 2\ln r)\{(\xi - x)\cos\beta + (\eta - y)\sin\beta\}$$

$$\frac{\partial^2 w^*(x,y;\xi,\eta)}{\partial x \partial \xi} = -\frac{1}{8\pi D}\left\{1 + 2\ln r + \frac{2(\xi - x)^2}{r^2}\right\}$$

$$\frac{\partial^2 w^*(x,y;\xi,\eta)}{\partial x \partial \eta} = -\frac{1}{4\pi D}\frac{(\xi - x)(\eta - y)}{r^2}$$

$$\frac{\partial^2 w^*(x,y;\xi,\eta)}{\partial y \partial \xi} = -\frac{1}{4\pi D}\frac{(\xi - x)(\eta - y)}{r^2}$$

$$\frac{\partial^2 w^*(x,y;\xi,\eta)}{\partial y \partial \eta} = -\frac{1}{8\pi D}\left\{1 + 2\ln r + \frac{2(\eta - y)^2}{r^2}\right\}$$

$$\frac{\partial^2 w^*(x,y;\xi,\eta)}{\partial n(x,y) \partial \xi} = -\frac{1}{8\pi D}\left[\left\{1 + 2\ln r + \frac{2(\xi - x)^2}{r^2}\right\}\cos\beta + \frac{2(\xi - x)(\eta - y)\sin\beta}{r^2}\right]$$

$$\frac{\partial^2 w^*(x,y;\xi,\eta)}{\partial n(x,y) \partial \eta} = -\frac{1}{8\pi D}\left[\left\{1 + 2\ln r + \frac{2(\eta - y)^2}{r^2}\right\}\sin\beta + \frac{2(\xi - x)(\eta - y)\cos\beta}{r^2}\right]$$

$$M_n^*(x,y;\xi,\eta) = -\frac{1}{8\pi}\left[1 + 3\nu + 2(1+\nu)\ln r + (1-\nu)\frac{2}{r^2}\right.$$

$$\left.\{(\xi - x)\cos\alpha + (\eta - y)\sin\alpha\}^2\right]$$

$$\frac{\partial M_n^*(x,y;\xi,\eta)}{\partial x} = -\frac{1}{8\pi}\left[\frac{-2(1+\nu)(\xi - x)}{r^2} + 2(1-\nu)\left[\frac{-2\cos\alpha}{r^2}\right.\right.$$

$$\left.\{(\xi - x)\cos\alpha + (\eta - y)\sin\alpha\} + \frac{2(\xi - x)}{r^4}\right.$$

$$\left.\left.\{(\xi - x)\cos\alpha + (\eta - y)\sin\alpha\}^2\right]\right]$$

$$\frac{\partial M_n^*(x,y;\xi,\eta)}{\partial y} = -\frac{1}{8\pi}\left[\frac{-2(1+\nu)(\eta - y)}{r^2} + 2(1-\nu)\left[\frac{-2\sin\alpha}{r^2}\right.\right.$$

$$\left.\{(\xi - x)\cos\alpha + (\eta - y)\sin\alpha\} + \frac{2(\eta - y)}{r^4}\right.$$

$$\left.\left.\{(\xi - x)\cos\alpha + (\eta - y)\sin\alpha\}^2\right]\right]$$

$$\frac{\partial M_n^*(x,y;\xi,\eta)}{\partial n(x,y)} = -\frac{1}{4\pi r^2}\Big[-(1+\nu)\{(\xi-x)\cos\beta+(\eta-y)\sin\beta\}$$

$$+2(1-\nu)\Big[-\cos(\beta-\alpha)\{(\xi-x)\cos\alpha+(\eta-y)\sin\alpha\}$$

$$+\frac{1}{r^2}\{(\xi-x)\cos\beta+(\eta-y)\sin\beta\}$$

$$\{(\xi-x)\cos\alpha+(\eta-y)\sin\alpha\}^2\Big]\Big]$$

$$V_n^*(x,y;\xi,\eta) = -\frac{1}{4\pi r^2}\Big[\{1+\nu+6(1-\nu)\sin^2\alpha\}(\xi-x)\cos\alpha$$

$$+(1-\nu)(2-8\sin^2\alpha)\frac{(\xi-x)^3\cos\alpha}{r^2}$$

$$+\{1+\nu+6(1-\nu)\cos^2\alpha\}(\eta-y)\sin\alpha$$

$$+(1-\nu)(2-8\cos^2\alpha)\frac{(\eta-y)^3\sin\alpha}{r^2}\Big]$$

$$\frac{\partial V_n^*(x,y;\xi,\eta)}{\partial x} = -\frac{1}{4\pi r^2}\Big[\{1+\nu+6(1-\nu)\sin^2\alpha\}\{-1+\frac{2(\xi-x)^2}{r^2}\}\cos\alpha$$

$$+(1-\nu)(2-8\sin^2\alpha)\{\frac{-3}{r^2}+\frac{4(\xi-x)^2}{r^4}\}(\xi-x)^2\cos\alpha$$

$$+\{1+\nu+6(1-\nu)\cos^2\alpha\}\frac{2(\xi-x)(\eta-y)\sin\alpha}{r^2}$$

$$+(1-\nu)(2-8\cos^2\alpha)\frac{4(\xi-x)(\eta-y)^3\sin\alpha}{r^4}\Big]$$

$$\frac{\partial V_n^*(x,y;\xi,\eta)}{\partial y} = -\frac{1}{4\pi r^2}\Big[\{1+\nu+6(1-\nu)\sin^2\alpha\}\frac{2(\xi-x)(\eta-y)\cos\alpha}{r^2}$$

$$+(1-\nu)(2-8\sin^2\alpha)\frac{4(\xi-x)^3(\eta-y)\cos\alpha}{r^4}$$

$$+\{1+\nu+6(1-\nu)\cos^2\alpha\}\{-1+\frac{2(\eta-y)^2}{r^2}\}\sin\alpha$$

$$+(1-\nu)(2-8\cos^2\alpha)\{\frac{-3}{r^2}+\frac{4(\eta-y)^2}{r^4}\}(\eta-y)^2\sin\alpha\Big]$$

$$\frac{\partial V_n^*(x,y;\xi,\eta)}{\partial n(x,y)} = -\frac{1}{4\pi r^2}\Big[\{n_x(1+\nu) + 6n_x n_y^2(1-\nu)\}$$

$$\Big[\cos\beta\Big\{-1 + \frac{2(\xi-x)^2}{r^2}\Big\} + \sin\beta\Big\{\frac{2(\xi-x)(\eta-y)}{r^2}\Big\}\Big]$$

$$+ \{2n_x(1-\nu) - 8n_x n_y^2(1-\nu)\}\Big[\cos\beta\Big\{\frac{-3}{r^2} + \frac{4(\xi-x)^2}{r^4}\Big\} + \sin\beta\Big\{\frac{4(\xi-x)^3(\eta-y)}{r^4}\Big\}\Big]$$

$$+ \{n_y(1+\nu) + 6n_y n_x^2(1-\nu)\}$$

$$\Big[\cos\beta\Big\{\frac{2(\xi-x)(\eta-y)}{r^2}\Big\} + \sin\beta\Big\{-1 + \frac{2(\eta-y)^2}{r^2}\Big\}\Big]$$

$$+ \{2n_y(1-\nu) - 8n_y n_x^2(1-\nu)\}$$

$$\Big[\cos\beta\Big\{\frac{4(\xi-x)(\eta-y)^3}{r^4}\Big\} + \sin\beta(\eta-y)^2\Big\{\frac{-3}{r^2} + \frac{4(\eta-y)^2}{r^4}\Big\}\Big]\Big]$$

where $n_x = \cos\alpha$, $n_y = \sin\alpha$.

$$R^{*}(x,y;\xi,\eta) = \frac{(1-\nu)[\{(\eta-y)^2-(\xi-x)^2\}(\sin\alpha_1\cos\alpha_1 - \sin\alpha_2\cos\alpha_2) + 2(\xi-x)(\eta-y)(\cos^2\alpha_1-\cos^2\alpha_2)]}{4\pi r^2}$$

$$\frac{\partial R^{*}(x,y;\xi,\eta)}{\partial x} = \frac{(1-\nu)[(\sin\alpha_1\cos\alpha_1 - \sin\alpha_2\cos\alpha_2)(\xi-x)\{1+\{(\eta-y)^2-(\xi-x)^2\}\}}{2\pi r^2 \qquad\qquad r^2}$$

$$+ (\cos^2\alpha_1-\cos^2\alpha_2)(\eta-y)\{-1+\frac{2(\xi-x)^2}{r^2}\}]$$

$$\frac{\partial R^{*}(x,y;\xi,\eta)}{\partial y} = \frac{(1-\nu)[(\sin\alpha_1\cos\alpha_1 - \sin\alpha_2\cos\alpha_2)(\eta-y)\{-1+\{(\eta-y)^2-(\xi-x)^2\}\}}{2\pi r^2 \qquad\qquad r^2}$$

$$+ (\cos^2\alpha_1-\cos^2\alpha_2)(\xi-x)\{-1+\frac{2(\eta-y)^2}{r^2}\}]$$

$$\frac{\partial R^{*}(x,y;\xi,\eta)}{\partial n(x,y)} = \frac{(1-\nu)}{2\pi r^{2}}\Big[(\sin\alpha_1\cos\alpha_1 - \sin\alpha_2\cos\alpha_2)\frac{2(\xi-x)}{r^{2}}$$

$$(\eta - y)\{(\eta - y)\cos\beta - (\xi - x)\sin\beta\}$$

$$+ (\cos^{2}\alpha_1 - \cos^{2}\alpha_2)\Big[(\eta - y)\cos\beta\{-1 + \frac{2(\xi - x)^{2}}{r^{2}}\}$$

$$+ (\xi - x)\sin\beta\{-1 + \frac{2(\eta - y)^{2}}{r^{2}}\}\Big]\Big]$$

APPENDIX C

```
C*************************************************************
C                                                            *
C         ANALYSIS OF RECTILINEAR FLAT PLATES                *
C             OF ARBITRARY PLAN FORMS BY                     *
C              BOUNDARY INTEGRAL METHOD                      *
C                                                            *
C                        BY                                  *
C                   Yuthana  Laokeaw                         *
C                                                            *
C*************************************************************
C
C  MAIN PROGRAM
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /LIO/ LR,LW,LC,LM
      COMMON /SIZE/ MAX
      COMMON A(40000)
C  MAXIMUM STORAGE CAN BE USED
      MAX = 40000
C  SET INPUT-OUTPUT LOGICAL UNITS
      OPEN (5,FILE='I.DAT')
      OPEN (6,FILE='O.TXT')
      CLOSE (6,STATUS='DELETE')
      OPEN (6,FILE='O.TXT')
      OPEN (7,FILE='M.TXT')
      CLOSE (7,STATUS='DELETE')
      OPEN (7,FILE='M.TXT')
      LW = 1
      LR = 5
      LC = 6
      LM = 7
      CALL BEGIN
      STOP
      END
C
C*************************************************************
C
      SUBROUTINE BEGIN
C
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*80 HEAD
      COMMON A(1)
      COMMON /LIO/ LR,LW,LC,LM
      COMMON /SIZE/ MAX
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
      COMMON /STIFF/ STA(200),STR(200,2)
C
      PI = 3.141592653589793
C
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
```

```
      WRITE (LM,*) ' '
C READ TITLE HEADING
      READ (LR,1) HEAD
      WRITE (LW,1) HEAD
      WRITE (LC,1) HEAD
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
C READ NUMBER OF SIDES, COLUMNS AND EXECUTION MODE
      READ (LR,*) KSIDE,NCOL,MODEX
      WRITE (LW,*) ' C O N T R O L   P A R A M E T E R S'
      WRITE (LC,*) ' C O N T R O L   P A R A M E T E R S'
      WRITE (LW,1000) KSIDE,NCOL,MODEX
      WRITE (LC,1000) KSIDE,NCOL,MODEX
C READ CO-ORDINATES OF VERTEXS, NO. OF ELEMENTS PER SIDE
      WRITE (LW,1010)
      WRITE (LC,1010)
      DO 10 I = 1,KSIDE
   10 READ (LR,*) N,CV(N,1),CV(N,2),NELEM(N)
      DO 15 N = 1,KSIDE
      WRITE (LW,1020) N,CV(N,1),CV(N,2),NELEM(N)
   15 WRITE (LC,1020) N,CV(N,1),CV(N,2),NELEM(N)
C
C READ INTERIOR SUPPORT DATA
      WRITE (LW,1030)
      WRITE (LC,1030)
      DO 20 I = 1,NCOL
  20  READ (LR,*) N,(CC(N,J),J=1,2),STA(N),(STR(N,J),J=1,2)
      DO 25 N = 1,NCOL
      WRITE (LW,1040) N,(CC(N,J),J=1,2),STA(N),(STR(N,J),J=1,2)
  25  WRITE (LC,1040) N,(CC(N,J),J=1,2),STA(N),(STR(N,J),J=1,2)
C
C READ PLATE THICKNESS AND MATERIAL PROPERTIES
      WRITE (LW,1050)
      WRITE (LC,1050)
      READ (LR,*) TH,E,PR
      WRITE (LW,1060) TH,E,PR
      WRITE (LC,1060) TH,E,PR
C
C READ LOADING DATA
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) ' L O A D I N G   D A T A'
      WRITE (LC,*) ' L O A D I N G   D A T A'
      READ (LR,*) NPL,NZ,NCM
      WRITE (LW,1070) NPL,NZ,NCM
      WRITE (LC,1070) NPL,NZ,NCM
      IF (NPL.EQ.0) GO TO 40
      WRITE (LW,*) ' C O N C E N T R A T E D   L O A D'
      WRITE (LC,*) ' C O N C E N T R A T E D   L O A D'
      WRITE (LW,1080)
      WRITE (LC,1080)
      DO 30 I = 1,NPL
 30   READ (LR,*) N,PL(N),(XP(N,J),J=1,2)
```

```
      DO 35 N = 1,NPL
      WRITE (LW,1090) N,PL(N),(XP(N,J),J=1,2)
 35   WRITE (LC,1090) N,PL(N),(XP(N,J),J=1,2)
 40   IF (NZ.EQ.0) GO TO 60
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) ' D I S T R I B U T E D   L O A D'
      WRITE (LC,*) ' D I S T R I B U T E D   L O A D'
      DO 50 I = 1,NZ
 50   READ (LR,*) N,INCRM(N),NS(N),U(1,N),U(2,N),CVL(1,1,N),
     @CVL(1,2,N),CVL(2,1,N),CVL(2,2,N),CVL(3,1,N),CVL(3,2,N),
     @CVL(4,1,N),CVL(4,2,N)
      WRITE (LW,1100)
      WRITE (LC,1100)
      DO 52 I = 1,NZ
      WRITE (LW,1105) I,INCRM(I),NS(I)
 52   WRITE (LC,1105) I,INCRM(I),NS(I)
      WRITE (LW,1110)
      WRITE (LC,1110)
      DO 55 I = 1,NZ
      WRITE (LW,1115) I,U(1,I),U(2,I),CVL(1,1,I),
     @CVL(1,2,I),CVL(2,1,I),CVL(2,2,I),CVL(3,1,I),CVL(3,2,I),
     @CVL(4,1,I),CVL(4,2,I)
 55   WRITE (LC,1115) I,U(1,I),U(2,I),CVL(1,1,I),
     @CVL(1,2,I),CVL(2,1,I),CVL(2,2,I),CVL(3,1,I),CVL(3,2,I),
     @CVL(4,1,I),CVL(4,2,I)
 60   IF (NCM.EQ.0) GO TO 65
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) ' C O N C E N T R A T E D   M O M E N T'
      WRITE (LC,*) ' C O N C E N T R A T E D   M O M E N T'
      WRITE (LW,2000)
      WRITE (LC,2000)
      DO 62 I = 1,NCM
 62   READ (LR,*) N,CM(N,1),CM(N,2),(CCM(N,J),J=1,2)
      DO 64 N = 1,NCM
      WRITE (LW,2010) N,CM(N,1),CM(N,2),(CCM(N,J),J=1,2)
 64   WRITE (LC,2010) N,CM(N,1),CM(N,2),(CCM(N,J),J=1,2)
C
C READ SOLUTION OUTPUT
 65   READ (LR,*) NDL
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) ' NUMBER OF SOLUTION LINES   =',NDL
      WRITE (LC,*) ' NUMBER OF SOLUTION LINES   =',NDL
      WRITE (LW,1120)
      WRITE (LC,1120)
      DO 70 I = 1,NDL
 70   READ (LR,*) N,CDLI(N,1),CDLI(N,2),CDLJ(N,1),CDLJ(N,2),NI(N)
      DO 75 N = 1,NDL
      WRITE (LW,1130) N,CDLI(N,1),CDLI(N,2),CDLJ(N,1),CDLJ(N,2),NI(N)
 75   WRITE (LC,1130) N,CDLI(N,1),CDLI(N,2),CDLJ(N,1),CDLJ(N,2),NI(N)
      NTEMP = 0
```

```
      DO 80 I = 1,KSIDE
 80   NTEMP = NTEMP + NELEM(I)
      NRS = 2*NTEMP + KSIDE + 3*NCOL
      N1 = 1
      N2 = N1 + NRS*NRS
      N3 = N2 + NRS
      NE = N3 + 505
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) ' NO. OF EQUATIONS   = ',NRS
      WRITE (LC,*) ' NO. OF EQUATIONS   = ',NRS
      WRITE (LW,*) ' REQUIRED STORAGES  = ',NE
      WRITE (LC,*) ' REQUIRED STORAGES  = ',NE
      IF (NE.GT.MAX) GO TO 999
      IF (MODEX.EQ.1) GO TO 99
C
C COMPUTE RIGIDITY OF PLATE (D)
      D = E*TH**3/(12.*(1.-PR**2))
      WRITE (LW,*) ' PLATE RIGIDITY (D) = ',D
      WRITE (LC,*) ' PLATE RIGIDITY (D) = ',D
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
C
C GENERATE COEFFICIENT MATRIX
      CALL GENER (A(N1),NRS)
      WRITE (LW,*) '*** GENERATING OF COEFFICIENT MATRIX COMPLETED ***'
C     CALL DISP2 (A(N1),NRS,NRS)
C
C GENERATE LOAD VECTOR
C
      CALL GENLD (A(N2))
      WRITE (LW,*) '*** GENERATING OF LOAD VECTOR COMPLETED ***'
C     CALL DISP2 (A(N2),NRS,1)
C
      CALL SLNPD (A(N1),A(N2),NRS)
      WRITE (LW,*) '**** SOLVING OF UNKNOWNS COMPLETED ****'
C     CALL DISP2 (A(N2),NRS,1)
C     CALL WTAPE2 (A(N2),NRS,1)
C
      DO 90 I = 1,NDL
      CALL DEFLEC (A(N2),A(N3),I,NI(I)+1)
      WRITE (LW,*) ' SOLUTION LINE NUMBER ',I
      WRITE (LC,*) ' SOLUTION LINE NUMBER ',I
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) '                    W          Mn          Mnt
     @    Qn          Vn'
      WRITE (LC,*) '                    W          Mn          Mnt
     @    Qn          Vn'
C     CALL DISP2 (A(N3),NI(I)+1,5)
      CALL WTAPE2 (A(N3),NI(I)+1,5)
 90   CONTINUE
      NN1 = N2 + 2*NTEMP + KSIDE
```

```
      NN2 = NN1 + NCOL
      NN3 = NN2 + NCOL
      WRITE (LW,*) ' '
      WRITE (LC,*) ' '
      WRITE (LW,*) ' S U P P O R T   D I S P L A C E M E N T S   A N D
     @ R E A C T I O N S'
      WRITE (LC,*) ' S U P P O R T   D I S P L A C E M E N T S   A N D
     @ R E A C T I O N S'
      WRITE (LW,2020)
      WRITE (LC,2020)
      DO 100 I = 1,NCOL
      II = I - 1
      WRITE (LW,2030) I,A(NN1+II),A(NN2+II),A(NN3+II),A(NN1+II)*STA(I),
     @A(NN3+II)*STR(I,1),A(NN2+II)*STR(I,2)
  100 WRITE (LC,2030) I,A(NN1+II),A(NN2+II),A(NN3+II),A(NN1+II)*STA(I),
     @A(NN3+II)*STR(I,1),A(NN2+II)*STR(I,2)
C
      RETURN
  999 NMORE = NE - MAX
      WRITE (LW,3010) NE,NMORE
   99 RETURN
    1 FORMAT (A80)
 1000 FORMAT (/5X,'NUMBER OF SIDES    =',I5,/5X,'NUMBER OF SUPPORTS =',
     @I5,/5X,     'SOLUTION MODE      =',I5,
     @/7X,'  EQ. 0  PROBLEM SOLUTION',/7X,'  EQ. 1  DATA CHECK')
 1010 FORMAT (//1X,'B O U N D A R Y   D A T A',//1X,
     @'VERTEX            COORDINATES            NUMBER OF',/1X,
     @'NUMBER          X                 Y          INTERVALS')
 1020 FORMAT (3X,I3,2(5X,F10.5),7X,I3)
 1030 FORMAT (//1X,'I N T E R I O R   S U P P O R T   D A T A',//1X,
     @'SUPPORT        COORDINATES            AXIAL        X-ROTAT.
     @ Y-ROTAT.',/1X,
     @'NUMBER        X           Y        STIFFNESS      STIFFNESS
     @ STIFFNESS')
 1040 FORMAT (3X,I4,2(3X,F9.4),3(3X,E12.5))
 1050 FORMAT (//1X,'G E O M E T R I C   A N D   M A T E R I A L   P R O
     @P E R T I E S',//1X,
     @'    PLATE         YOUNG*S        POISSON*S',/1X,
     @'  THICKNESS       MODULUS          RATIO')
 1060 FORMAT (1X,F10.5,5X,E12.5,5X,F10.5)
 1070 FORMAT (/1X,' TOTAL NUMBER OF CONCENTRATED LOADS  = ',I5,/1X,
     @' TOTAL NUMBER OF ZONES SUBJECTED TO DISTRIBUTED LOAD  = ',I5,
     @/1X,' TOTAL NUMBER OF CONCENTRATED MOMENTS  =',I5,/)
 1080 FORMAT (/1X,' LOAD          Z-AXIS              COORDINATES',/1X,
     @          'NUMBER        FORCE              X            Y')
 1090 FORMAT (2X,I4,5X,E12.5,5X,F10.5,3X,F10.5)
 1100 FORMAT (/1X,' ZONE    DIVIDING    NUMBER',/1X,
     @          'NUMBER   DIRECTION  OF STRIPS')
 1105 FORMAT (3X,I3,6X,I3,8X,I4)
 1110 FORMAT (/1X,' ZONE              INTENSITY
     @                    COORDINATES',/1X,
     @          'NUMBER    1ST VERTEX     3RD VERTEX     1ST-X      1S
     @T-Y      2ND-X      2ND-Y      3RD-X      3RD-Y      4TH-X      4T
     @H-Y')
 1115 FORMAT (3X,I3,2(3X,E12.5),8(2X,F9.4))
```

```
 1120 FORMAT (//1X,'S O L U T I O N   O U T P U T',//1X,
     @' LINE                    COORDINATES                          NUMBER OF
     @',/1X,
     @'NUMBER      I-X          I-Y          J-X          J-Y      INTERVALS
     @')
 1130 FORMAT (2X,I4,4(3X,F9.4),5X,I4)
 2000 FORMAT (/1X,'MOMENT       X-AXIS           Y-AXIS              COO
     @RDINATES',/1X,
     @            'NUMBER       MOMENT           MOMENT              X
     @         Y')
 2010 FORMAT (2X,I4,5X,E12.5,5X,E12.5,5X,F10.5,3X,F10.5)
 2020 FORMAT (/1X,'  SUPPORT   DEFLECTION       SLOPE          SLOPE
     @     REACTION       MOMENT          MOMENT',/1X,
     @'    NO.                       W.R.T. X       W.R.T. Y
     @     ABOUT X-AXIS   ABOUT Y-AXIS',/)
 2030 FORMAT (2X,I6,6(3X,E12.5))
 3010 FORMAT (/5X,'REQUIRED STORAGES =',I6,' WHICH',I6,' EXCEEDED')
      END
C
C*************************************************************
C
      SUBROUTINE GENER (S,NRS)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION S(NRS,1)
      COMMON /LIO/ LR,LW,LC,LM
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
      COMMON /STIFF/ STA(200),STR(200,2)
C
      CALL ELEML
      KI = 0
      KI1 = NTEMP
C
      DO 100 K = 1,KSIDE
       DO 90 I = 1,NELEM(K)
       KI = KI + 1
       KI1 = KI1 + 1
       LJ = 0
        DO 20 L = 1,KSIDE
         DO 10 J = 1,NELEM(L)
          LJ = LJ + 1
          CALL CALR1 (K,I,L,J,R,BETA,ALPHA,DX1,DY1,DX2,DY2)
          IF (R.LE.0.0) THEN
          S(KI,LJ) = -EL(L)*(1.-PR-2.*(1.+PR)*DLOG(EL(L)/2.))/(8.*PI)
          S(KI1,LJ) = 0.5
          ELSE
          CALL TRANF (DX2,DY2,ALPHA,AN,T)
          S(KI,LJ) = -FUNCM (AN,T,K,L)
          S(KI1,LJ) = - DMN (AN,T,BETA,ALPHA,K,L)
          CALL TRANF (DX1,DY1,ALPHA,AN,T)
          S(KI,LJ) = S(KI,LJ) + FUNCM (AN,T,K,L)
          S(KI1,LJ) = S(KI1,LJ) + DMN (AN,T,BETA,ALPHA,K,L)
          END IF
```

```
   10    CONTINUE
   20   CONTINUE
C
        DO 40 L = 1,KSIDE
         DO 30 J = 1,NELEM(L)
          LJ = LJ + 1
          CALL CALR1 (K,I,L,J,R,BETA,ALPHA,DX1,DY1,DX2,DY2)
          IF (R.LE.0.0) THEN
          S(KI,LJ) = 0.5
          S(KI1,LJ) = -(1.+PR)/(PI*EL(L))
          ELSE
          CALL TRANF (DX2,DY2,ALPHA,AN,T)
          S(KI,LJ) = FUNCV (AN,T,K,L)
          S(KI1,LJ) = DVN (AN,T,BETA,ALPHA)
          CALL TRANF (DX1,DY1,ALPHA,AN,T)
          S(KI,LJ) = S(KI,LJ) - FUNCV (AN,T,K,L)
          S(KI1,LJ) = S(KI1,LJ) - DVN (AN,T,BETA,ALPHA)
          END IF
   30    CONTINUE
   40   CONTINUE
C
        DO 50 L = 1,KSIDE
          LJ = LJ + 1
          CALL CALR (K,I,L,0,R,BETA,ALPHA,DX,DY)
          CALL ANOR (L,ALPHA1,ALPHA2)
          IF (R.LE.0.0) THEN
          S(KI,LJ) = 0.0
          S(KI1,LJ) = 0.0
          ELSE
          S(KI,LJ) = FUNCR (R,DX,DY,ALPHA1,ALPHA2)
          S(KI1,LJ) = DRN (R,DX,DY,BETA,ALPHA1,ALPHA2)
          END IF
   50   CONTINUE
C
        DO 60 L = 1,NCOL
          LJ  = LJ + 1
          LJ1 = LJ + NCOL
          LJ2 = LJ1 + NCOL
          CALL CALR (K,I,0,L,R,BETA,ALPHA,DX,DY)
          S(KI,LJ)  = FUNCW(R)*STA(L)
          S(KI,LJ1) = DWXI (R,DX)*STR(L,2)
          S(KI,LJ2) = DWETA (R,DY)*STR(L,1)
          S(KI1,LJ)  = DWN (R,DX,DY,BETA)*STA(L)
          S(KI1,LJ1) = DWXIN (R,DX,DY,BETA)*STR(L,2)
          S(KI1,LJ2) = DWETAN (R,DX,DY,BETA)*STR(L,1)
   60    CONTINUE
   90   CONTINUE
  100  CONTINUE
C
       KI = 2*NTEMP
       DO 300 K = 1,KSIDE
        KI = KI + 1
        LJ = 0
         DO 220 L = 1,KSIDE
          DO 210 J = 1,NELEM(L)
```

```
          LJ = LJ + 1
          CALL CALR1 (K,0,L,J,R,BETA,ALPHA,DX1,DY1,DX2,DY2)
          ICODE = 0
          LL = L
          IF (K.EQ.1.AND.L.EQ.KSIDE) LL = K
          IF (L.EQ.K-1) LL = K
           IF (K.EQ.L.AND.J.EQ.1) ICODE = 1
           IF (L.EQ.K-1.AND.J.EQ.NELEM(L)) ICODE = 1
           IF (L.EQ.KSIDE.AND.K.EQ.1.AND.J.EQ.NELEM(L)) ICODE = 1
          IF (ICODE.EQ.1) THEN
          S(KI,LJ) = -EL(L)*(1.-PR-2.*(1.+PR)*DLOG(EL(L)))/(8.*PI)
          ELSE
          CALL TRANF (DX2,DY2,ALPHA,AN,T)
          S(KI,LJ) = -FUNCM (AN,T,K,LL)
          CALL TRANF (DX1,DY1,ALPHA,AN,T)
          S(KI,LJ) = S(KI,LJ) + FUNCM (AN,T,K,LL)
          END IF
  210    CONTINUE
  220   CONTINUE
C
        DO 240 L = 1,KSIDE
         DO 230 J = 1,NELEM(L)
          LJ = LJ + 1
          CALL CALR1 (K,0,L,J,R,BETA,ALPHA,DX1,DY1,DX2,DY2)
          LL = L
          IF (K.EQ.1.AND.L.EQ.KSIDE) LL = K
          IF (L.EQ.K-1) LL = K
          CALL TRANF (DX2,DY2,ALPHA,AN,T)
          S(KI,LJ) = FUNCV (AN,T,K,LL)
          CALL TRANF (DX1,DY1,ALPHA,AN,T)
          S(KI,LJ) = S(KI,LJ) - FUNCV (AN,T,K,LL)
  230    CONTINUE
  240   CONTINUE
C
        DO 250 L = 1,KSIDE
          LJ = LJ + 1
          CALL CALR (K,0,L,0,R,BETA,ALPHA,DX,DY)
          CALL ANOR (L,ALPHA1,ALPHA2)
          IF (R.LE.0.0) THEN
          S(KI,LJ) =   0.00 + AIN (ALPHA1,ALPHA2)/(2.*PI)
          ELSE
          S(KI,LJ) = FUNCR (R,DX,DY,ALPHA1,ALPHA2)
          END IF
  250   CONTINUE
C
        DO 260 L = 1,NCOL
          LJ = LJ + 1
          LJ1 = LJ + NCOL
          LJ2 = LJ1 + NCOL
          CALL CALR (K,0,0,L,R,BETA,ALPHA,DX,DY)
          S(KI,LJ)  = FUNCW(R)*STA(L)
          S(KI,LJ1) = DWXI (R,DX)*STR(L,2)
          S(KI,LJ2) = DWETA (R,DY)*STR(L,1)
  260   CONTINUE
  300 CONTINUE
```

```
C
      KI1 = 2*NTEMP + KSIDE + NCOL
      KI2 = KI1 + NCOL
      DO 400 K = 1,NCOL
       KI = KI + 1
       KI1 = KI1 + 1
       KI2 = KI2 + 1
       LJ = 0
        DO 320 L = 1,KSIDE
         DO 310 J = 1,NELEM(L)
          LJ = LJ + 1
          CALL CALR1 (0,K,L,J,R,BETA,ALPHA,DX1,DY1,DX2,DY2)
          CALL TRANF (DX2,DY2,ALPHA,AN,T)
          S(KI,LJ) = -FUNCM (AN,T,K,K-1)
          BETA = 0.0
          S(KI1,LJ) = - DMN (AN,T,BETA,ALPHA,K,K-1)
          BETA = PI/2.0
          S(KI2,LJ) = - DMN (AN,T,BETA,ALPHA,K,K-1)
          CALL TRANF (DX1,DY1,ALPHA,AN,T)
          S(KI,LJ) = S(KI,LJ) + FUNCM (AN,T,K,K-1)
          BETA = 0.0
          S(KI1,LJ) = S(KI1,LJ) + DMN (AN,T,BETA,ALPHA,K,K-1)
          BETA = PI/2.0
          S(KI2,LJ) = S(KI2,LJ) + DMN (AN,T,BETA,ALPHA,K,K-1)
  310    CONTINUE
  320   CONTINUE
C
        DO 340 L = 1,KSIDE
         DO 330 J = 1,NELEM(L)
          LJ = LJ + 1
          CALL CALR1 (0,K,L,J,R,BETA,ALPHA,DX1,DY1,DX2,DY2)
          CALL TRANF (DX2,DY2,ALPHA,AN,T)
          S(KI,LJ) = FUNCV (AN,T,K,K-1)
          BETA = 0.0
          S(KI1,LJ) = DVN (AN,T,BETA,ALPHA)
          BETA = PI/2.0
          S(KI2,LJ) = DVN (AN,T,BETA,ALPHA)
          CALL TRANF (DX1,DY1,ALPHA,AN,T)
          S(KI,LJ) = S(KI,LJ) - FUNCV (AN,T,K,K-1)
          BETA = 0.0
          S(KI1,LJ) = S(KI1,LJ) - DVN (AN,T,BETA,ALPHA)
          BETA = PI/2.0
          S(KI2,LJ) = S(KI2,LJ) - DVN (AN,T,BETA,ALPHA)
  330    CONTINUE
  340   CONTINUE
C
        DO 350 L = 1,KSIDE
          LJ = LJ + 1
          CALL CALR (0,K,L,0,R,BETA,ALPHA,DX,DY)
          CALL ANOR (L,ALPHA1,ALPHA2)
          S(KI,LJ) = FUNCR (R,DX,DY,ALPHA1,ALPHA2)
          S(KI1,LJ) = DRX (R,DX,DY,ALPHA1,ALPHA2)
          S(KI2,LJ) = DRY (R,DX,DY,ALPHA1,ALPHA2)
  350   CONTINUE
C
```

```
      DO 360 L = 1,NCOL
        LJ = LJ + 1
        LJ1 = LJ + NCOL
        LJ2 = LJ1 + NCOL
        CALL CALR (0,K,0,L,R,BETA,ALPHA,DX,DY)
        IF (R.LE.0.0) THEN
        S(KI,LJ)  = 1.0
        S(KI,LJ1) = 0.0
        S(KI,LJ2) = 0.0
        S(KI1,LJ)  = 0.0
        S(KI1,LJ1) = 1.0
        S(KI1,LJ2) = 0.0
        S(KI2,LJ)  = 0.0
        S(KI2,LJ1) = 0.0
        S(KI2,LJ2) = 1.0
        ELSE
        S(KI,LJ)  = FUNCW(R)*STA(L)
        S(KI,LJ1) = DWXI (R,DX)*STR(L,2)
        S(KI,LJ2) = DWETA (R,DY)*STR(L,1)
        S(KI1,LJ)  = DWX (R,DX)*STA(L)
        S(KI1,LJ1) = DWXIX (R,DX)*STR(L,2)
        S(KI1,LJ2) = DWETAX (R,DX,DY)*STR(L,1)
        S(KI2,LJ)  = DWY (R,DY)*STA(L)
        S(KI2,LJ1) = DWXIY (R,DX,DY)*STR(L,2)
        S(KI2,LJ2) = DWETAY (R,DY)*STR(L,1)
        END IF
  360   CONTINUE
  400 CONTINUE
C
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE GENLD (Q)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Q(1)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      NRS = 2*NTEMP + KSIDE + 3*NCOL
      DO 50 I = 1,NRS
   50 Q(I) = 0.0
      IF (NPL.EQ.0) GO TO 20
      DO 10 I = 1,NPL
   10 CALL GENLD1 (Q,PL(I),XP(I,1),XP(I,2))
   20 IF (NZ.EQ.0) GO TO 30
      DO 500 I = 1,NZ
       ANS = DBLE(NS(I))
       II = INCRM(I) + 1
       SW = DABS((CVL(1,II,I) - CVL(4,II,I))/ANS)
```

```
      DO 400 J = 1,NS(I)
      CALL COORL (I,J,XI1,ETA1,XI2,ETA2,UU)
      UL = UU*SW
      CALL GENLD2 (Q,UL,XI1,ETA1,XI2,ETA2,I)
  400 CONTINUE
  500 CONTINUE
   30 IF (NCM.EQ.0) GO TO 40
      DO 600 I = 1,NCM
  600 CALL GENLD3 (Q,CM(I,1),CM(I,2),CCM(I,1),CCM(I,2))
   40 RETURN
      END
C
C*************************************************************
C
      SUBROUTINE SLNPD (A,B,N)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(N,1),B(1)
      COMMON /LIO/ LR,LW,LC,LM
C
      N1 = N - 1
      DO 100 K = 1, N1
          WRITE (LW,5000) K
 5000     FORMAT (I5)
        K1 = K + 1
        C = A(K,K)
        IF (DABS(C)-0.000001) 1,1,3
    1   DO 7 J = K1, N
C
          IF (DABS(A(J,K)-0.000001)) 7,7,5
    5     DO 6 L = K, N
            C = A(K,L)
            A(K,L) = A(J,L)
            A(J,L) = C
    6     CONTINUE
          C = B(K)
          B(K) = B(J)
          B(J) = C
          C = A(K,K)
          GO TO 3
    7   CONTINUE
    8   WRITE (LW,2002) K
        D = 0.0
        GO TO 300
C
    3   C = A(K,K)
        DO 4 J = K1, N
          A(K,J) = A(K,J)/C
    4   CONTINUE
          B(K) = B(K)/C
C
        DO 10 I = K1, N
          C = A(I,K)
          DO 9 J = K1, N
            A(I,J) = A(I,J) - C*A(K,J)
```

```
    9     CONTINUE
          B(I) = B(I) - C*B(K)
   10   CONTINUE
  100 CONTINUE
C
      IF (DABS(A(N,N))-0.000001) 101,101,102
  101 WRITE (LW,2002) K
      D = 0.0
      GO TO 300
  102 B(N) = B(N)/A(N,N)
C
      DO 200 L = 1, N1
        K = N - L
        K1 = K + 1
        DO 200 J = K1, N
          B(K) = B(K) - A(K,J)*B(J)
  200 CONTINUE
  300 RETURN
 2002 FORMAT (/10X,'**** SINGULAR IN ROW',I5)
      END
C
C*************************************************************
C
      SUBROUTINE DEFLEC (Q,WP,I,N)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Q(1),WP(N,1)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
      COMMON /STIFF/ STA(200),STR(200,2)
C
      X1 = CDLI(I,1)
      Y1 = CDLI(I,2)
      X2 = CDLJ(I,1)
      Y2 = CDLJ(I,2)
      BETA = ANGLE1 (X1,Y1,X2,Y2)
C
      DO 200 M = 1,NI(I)+1
        AM = DBLE(M)
        ANI = DBLE(NI(I))
         X = (AM-1.0)*(X2-X1)/ANI + X1
         Y = (AM-1.0)*(Y2-Y1)/ANI + Y1
C
         W = 0.0
         W1 = 0.0
         W2 = 0.0
         W3 = 0.0
         W4 = 0.0
         KI1 = 0
         KI2 = NTEMP
         K3  = 2*KI2
         K4  = K3 + KSIDE
```

```
      K5  = K4 + NCOL
      K6  = K5 + NCOL
      DO 90 L = 1,KSIDE
       K3 = K3 + 1
       DX = CV(L,1) - X
       DY = CV(L,2) - Y
       R  = DSQRT (DX**2 + DY**2)
       IF (R.LE.0.0) GO TO 40
       CALL ANOR (L,ALPHA1,ALPHA2)
       W = W - Q(K3)*FUNCR (R,DX,DY,ALPHA1,ALPHA2)
       W1 = W1 - Q(K3)*CMN (R,DX,DY,BETA,ALPHA1,ALPHA2)
       W2 = W2 - Q(K3)*CMNT (R,DX,DY,BETA,ALPHA1,ALPHA2)
       W3 = W3 - Q(K3)*CQN (R,DX,DY,BETA,ALPHA1,ALPHA2)
       W4 = W4 - Q(K3)*CVN (R,DX,DY,BETA,ALPHA1,ALPHA2)
40     DO 50 J = 1,NELEM(L)
        KI1 = KI1 + 1
        KI2 = KI2 + 1
        CALL PLOAD (X,Y,L,J,SM1,SM2,2)
        W = W + Q(KI1)*SM1
        W = W - Q(KI2)*SM2
        CALL PLOAD1 (X,Y,L,J,SM1,SM2,SM3,SM4,SM5,SM6,SM7,SM8
     @  ,BETA)
        W1 = W1 + Q(KI1)*SM1
        W1 = W1 - Q(KI2)*SM2
        W2 = W2 + Q(KI1)*SM3
        W2 = W2 - Q(KI2)*SM4
        W3 = W3 + Q(KI1)*SM5
        W3 = W3 - Q(KI2)*SM6
        W4 = W4 + Q(KI1)*SM7
        W4 = W4 - Q(KI2)*SM8
50     CONTINUE
90    CONTINUE
      DO 300 L = 1,NCOL
       K4 = K4 + 1
       K5 = K5 + 1
       K6 = K6 + 1
       DX = CC(L,1) - X
       DY = CC(L,2) - Y
       R = DSQRT (DX**2 + DY**2)
       IF (R.LE.0.0) GO TO 300
       W = W - Q(K4)*FUNCW (R)*STA(L)
       W = W - Q(K5)*DWXI (R,DX)*STR(L,2)
       W = W - Q(K6)*DWETA (R,DY)*STR(L,1)
       W1 = W1 - Q(K4)*DDMN (R,DX,DY,BETA)*STA(L)
       W1 = W1 - Q(K5)*EMN (R,DX,DY,BETA)*STR(L,2)
       W1 = W1 - Q(K6)*FMN (R,DX,DY,BETA)*STR(L,1)
       W2 = W2 - Q(K4)*DDMNT (R,DX,DY,BETA)*STA(L)
       W2 = W2 - Q(K5)*EMNT (R,DX,DY,BETA)*STR(L,2)
       W2 = W2 - Q(K6)*FMNT (R,DX,DY,BETA)*STR(L,1)
       W3 = W3 - Q(K4)*DDQN (R,DX,DY,BETA)*STA(L)
       W3 = W3 - Q(K5)*EQN (R,DX,DY,BETA)*STR(L,2)
       W3 = W3 - Q(K6)*FQN (R,DX,DY,BETA)*STR(L,1)
       W4 = W4 - Q(K4)*DDVN (R,DX,DY,BETA)*STA(L)
       W4 = W4 - Q(K5)*EVN (R,DX,DY,BETA)*STR(L,2)
       W4 = W4 - Q(K6)*FVN (R,DX,DY,BETA)*STR(L,1)
```

```
300     CONTINUE
        IF (NPL.EQ.0) GO TO 450
        DO 400 II = 1,NPL
        DX = XP(II,1) - X
        DY = XP(II,2) - Y
        R = DSQRT (DX*DX + DY*DY)
        IF (R.LE.0.0) GO TO 400
        W = W + FUNCW (R)*PL(II)
        W1 = W1 + DDMN (R,DX,DY,BETA)*PL(II)
        W2 = W2 + DDMNT (R,DX,DY,BETA)*PL(II)
        W3 = W3 + DDQN (R,DX,DY,BETA)*PL(II)
        W4 = W4 + DDVN (R,DX,DY,BETA)*PL(II)
400     CONTINUE
450     IF (NZ.EQ.0) GO TO 650
        DO 600 IZ = 1,NZ
         ANS = DBLE(NS(IZ))
         II  = INCRM(IZ) + 1
         SW  = DABS ((CVL(1,II,IZ) - CVL(4,II,IZ))/ANS)
          DO 500 JS = 1,NS(IZ)
            CALL COORL (IZ,JS,XI1,ETA1,XI2,ETA2,UU)
            UL = UU*SW
            W = W + UL*WINT (X,Y,XI2,ETA2,IZ)
            W = W - UL*WINT (X,Y,XI1,ETA1,IZ)
            W1 = W1 + UL*DIMN (X,Y,XI2,ETA2,BETA,IZ)
            W1 = W1 - UL*DIMN (X,Y,XI1,ETA1,BETA,IZ)
            W2 = W2 + UL*DIMNT (X,Y,XI2,ETA2,BETA,IZ)
            W2 = W2 - UL*DIMNT (X,Y,XI1,ETA1,BETA,IZ)
            W3 = W3 + UL*DIQN (X,Y,XI2,ETA2,BETA,IZ)
            W3 = W3 - UL*DIQN (X,Y,XI1,ETA1,BETA,IZ)
            W4 = W4 + UL*DIVN (X,Y,XI2,ETA2,BETA,IZ)
            W4 = W4 - UL*DIVN (X,Y,XI1,ETA1,BETA,IZ)
500       CONTINUE
600     CONTINUE
650     IF (NCM.EQ.0) GO TO 95
        DO 700 II = 1,NCM
         DX = CCM(II,1) - X
         DY = CCM(II,2) - Y
         R  = DSQRT (DX*DX + DY*DY)
         IF (R.LE.0.0) GO TO 700
           W  = W  + CM(II,1)*DWETA (R,DY)
           W  = W  - CM(II,2)*DWXI (R,DX)
           W1 = W1 + CM(II,1)*FMN (R,DX,DY,BETA)
           W1 = W1 - CM(II,2)*EMN (R,DX,DY,BETA)
           W2 = W2 + CM(II,1)*FMNT (R,DX,DY,BETA)
           W2 = W2 - CM(II,2)*EMNT (R,DX,DY,BETA)
           W3 = W3 + CM(II,1)*FQN (R,DX,DY,BETA)
           W3 = W3 - CM(II,2)*EQN (R,DX,DY,BETA)
           W4 = W4 + CM(II,1)*FVN (R,DX,DY,BETA)
           W4 = W4 - CM(II,2)*EVN (R,DX,DY,BETA)
700     CONTINUE
 95     WP(M,1) = W
        WP(M,2) = W1
        WP(M,3) = W2
        WP(M,4) = W3
        WP(M,5) = W4
```

```
  200 CONTINUE
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE GENLD1 (Q,CK,X,Y)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Q(1)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
       KI1 = 0
       KI2 = NTEMP
       K3  = 2*KI2
       K4  = K3 + KSIDE
       K5  = K4 + NCOL
       K6  = K5 + NCOL
      DO 200 K = 1,KSIDE
       K3 = K3 + 1
       R  = DSQRT ((CV(K,1)-X)**2+(CV(K,2)-Y)**2)
       IF (R.LE.0.0) THEN
        Q(K3) = Q(K3) + 0.0
       ELSE
        Q(K3) = Q(K3) + FUNCW (R)*CK
       END IF
       DO 100 I = 1,NELEM(K)
         KI1 = KI1 + 1
         KI2 = KI2 + 1
         CALL PLOAD (X,Y,K,I,SM1,SM2,1)
         Q(KI1) = Q(KI1) + SM1*CK
         Q(KI2) = Q(KI2) + SM2*CK
  100  CONTINUE
  200 CONTINUE
      DO 300 K = 1,NCOL
        K4 = K4 + 1
        K5 = K5 + 1
        K6 = K6 + 1
        R  = DSQRT ((CC(K,1)-X)**2+(CC(K,2)-Y)**2)
        DX = X - CC(K,1)
        DY = Y - CC(K,2)
        IF (R.LE.0.0) THEN
         Q(K4) = Q(K4) + 0.0
         Q(K5) = Q(K5) + 0.0
         Q(K6) = Q(K6) + 0.0
        ELSE
         Q(K4) = Q(K4) + FUNCW (R)*CK
         Q(K5) = Q(K5) + DWX (R,DX)*CK
         Q(K6) = Q(K6) + DWY (R,DY)*CK
        END IF
  300 CONTINUE
```

```
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE GENLD2 (Q,CK,XI1,ETA1,XI2,ETA2,IZ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Q(1)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
       KI1 = 0
       KI2 = NTEMP
       K3  = 2*KI2
       K4  = K3 + KSIDE
       K5  = K4 + NCOL
       K6  = K5 + NCOL
C
      DO 200 K = 1,KSIDE
       K3 = K3 + 1
       Q(K3) = Q(K3) + CK*WINT (CV(K,1),CV(K,2),XI2,ETA2,IZ)
       Q(K3) = Q(K3) - CK*WINT (CV(K,1),CV(K,2),XI1,ETA1,IZ)
      DO 100 I = 1,NELEM(K)
         KI1 = KI1 + 1
         KI2 = KI2 + 1
         CALL PLOAD2 (XI1,ETA1,XI2,ETA2,K,I,SM1,SM2,IZ)
         Q(KI1) = Q(KI1) + SM1*CK
         Q(KI2) = Q(KI2) + SM2*CK
  100  CONTINUE
  200 CONTINUE
      DO 300 K = 1,NCOL
        K4 = K4 + 1
        K5 = K5 + 1
        K6 = K6 + 1
        Q(K4) = Q(K4) + CK*WINT (CC(K,1),CC(K,2),XI2,ETA2,IZ)
        Q(K5) = Q(K5) + CK*DWXINT (CC(K,1),CC(K,2),XI2,ETA2,IZ)
        Q(K6) = Q(K6) + CK*DWYINT (CC(K,1),CC(K,2),XI2,ETA2,IZ)
        Q(K4) = Q(K4) - CK*WINT (CC(K,1),CC(K,2),XI1,ETA1,IZ)
        Q(K5) = Q(K5) - CK*DWXINT (CC(K,1),CC(K,2),XI1,ETA1,IZ)
        Q(K6) = Q(K6) - CK*DWYINT (CC(K,1),CC(K,2),XI1,ETA1,IZ)
  300 CONTINUE
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE GENLD3 (Q,CKX,CKY,X,Y)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Q(1)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
```

```
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
       KI1 = 0
       KI2 = NTEMP
       K3  = 2*KI2
       K4  = K3 + KSIDE
       K5  = K4 + NCOL
       K6  = K5 + NCOL
C
      DO 200 K = 1,KSIDE
       K3 = K3 + 1
       DX = X - CV(K,1)
       DY = Y - CV(K,2)
       R  = DSQRT (DX*DX + DY*DY)
       IF (R.LE.0.0) THEN
        Q(K3) = Q(K3) + 0.0
       ELSE
        Q(K3) = Q(K3) + CKX*DWETA (R,DY)
        Q(K3) = Q(K3) - CKY*DWXI (R,DX)
       END IF
       DO 100 I = 1,NELEM(K)
         KI1 = KI1 + 1
         KI2 = KI2 + 1
         CALL PLOAD3 (X,Y,K,I,SM1,SM2,SM3,SM4)
         Q(KI1) = Q(KI1) + SM1*CKX
         Q(KI1) = Q(KI1) - SM2*CKY
         Q(KI2) = Q(KI2) + SM3*CKX
         Q(KI2) = Q(KI2) - SM4*CKY
  100  CONTINUE
  200 CONTINUE
      DO 300 K = 1,NCOL
        K4 = K4 + 1
        K5 = K5 + 1
        K6 = K6 + 1
        DX = X - CC(K,1)
        DY = Y - CC(K,2)
        R  = DSQRT (DX*DX + DY*DY)
        IF (R.LE.0.0) THEN
         Q(K4) = Q(K4) + 0.0
         Q(K5) = Q(K5) + 0.0
         Q(K6) = Q(K6) + 0.0
        ELSE
         Q(K4) = Q(K4) + CKX*DWETA (R,DY)
         Q(K4) = Q(K4) - CKY*DWXI (R,DX)
         Q(K5) = Q(K5) + CKX*DWETAX (R,DX,DY)
         Q(K5) = Q(K5) - CKY*DWXIX (R,DX)
         Q(K6) = Q(K6) + CKX*DWETAY (R,DY)
         Q(K6) = Q(K6) - CKY*DWXIY (R,DX,DY)
        END IF
  300 CONTINUE
      RETURN
      END
```

```
C
C*************************************************************
C
      SUBROUTINE PLOAD (X,Y,K,I,SM1,SM2,IWAY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      CALL COOR (K,I,XI,ETA)
      RR =  DSQRT((XI-X)**2+(ETA-Y)**2)
        GO TO (2,4), IWAY
 2    DDX = X - XI
      DDY = Y - ETA
        IF (RR.LE.0.0) THEN
        SM1 = 0.0
        SM2 = 0.0
        ELSE
        SM1 = FUNCW (RR)
        CALL CALR (K,I,0,0,R,BETA,ALPHA,DX,DY)
        SM2 = DWN (RR,DDX,DDY,BETA)
        END IF
        RETURN
 4    CALL COOR1 (K,I,XI1,ETA1,XI2,ETA2)
        DDX1 = XI1 - X
        DDY1 = ETA1 - Y
        DDX2 = XI2 - X
        DDY2 = ETA2 - Y
        CALL CALR (0,0,K,I,R,BETA,ALPHA,DX,DY)
          CALL TRANF (DDX2,DDY2,ALPHA,AN,T)
          SM1 = FUNCM (AN,T,K,K-1)
          SM2 = FUNCV (AN,T,K,K-1)
          CALL TRANF (DDX1,DDY1,ALPHA,AN,T)
          SM1 = SM1 - FUNCM (AN,T,K,K-1)
          SM2 = SM2 - FUNCV (AN,T,K,K-1)
        RETURN
      END
C
C*************************************************************
C
      SUBROUTINE PLOAD1 (X,Y,K,I,SM1,SM2,SM3,SM4,SM5,SM6,SM7,SM8
     *,BETA)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      CALL COOR1 (K,I,XI1,ETA1,XI2,ETA2)
      DDX1 = XI1 - X
      DDY1 = ETA1 - Y
      DDX2 = XI2 - X
      DDY2 = ETA2 - Y
      CALL CALR (0,0,K,I,R,B,ALPHA,DX,DY)
```

```
      CALL TRANF (DDX2,DDY2,ALPHA,AN,T)
       SM1 = AMN (AN,T,BETA-ALPHA)
       SM2 = BMN (AN,T,BETA-ALPHA)
       SM3 = AMNT (AN,T,BETA-ALPHA)
       SM4 = BMNT (AN,T,BETA-ALPHA)
       SM5 = AQN (AN,T,BETA-ALPHA)
       SM6 = BQN (AN,T,BETA-ALPHA)
       SM7 = AVN (AN,T,BETA-ALPHA)
       SM8 = BVN (AN,T,BETA-ALPHA)
      CALL TRANF (DDX1,DDY1,ALPHA,AN,T)
       SM1 = SM1 - AMN (AN,T,BETA-ALPHA)
       SM2 = SM2 - BMN (AN,T,BETA-ALPHA)
       SM3 = SM3 - AMNT (AN,T,BETA-ALPHA)
       SM4 = SM4 - BMNT (AN,T,BETA-ALPHA)
       SM5 = SM5 - AQN (AN,T,BETA-ALPHA)
       SM6 = SM6 - BQN (AN,T,BETA-ALPHA)
       SM7 = SM7 - AVN (AN,T,BETA-ALPHA)
       SM8 = SM8 - BVN (AN,T,BETA-ALPHA)
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE PLOAD2 (XI1,ETA1,XI2,ETA2,K,I,SM1,SM2,IZ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      CALL COOR (K,I,X,Y)
      CALL CALR (K,I,0,0,R,BETA,ALPHA,DX,DY)
      SM1 = WINT (X,Y,XI2,ETA2,IZ)
      SM2 = DWNINT (X,Y,XI2,ETA2,BETA,IZ)
      SM1 = SM1 - WINT (X,Y,XI1,ETA1,IZ)
      SM2 = SM2 - DWNINT (X,Y,XI1,ETA1,BETA,IZ)
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE PLOAD3 (X,Y,K,I,SM1,SM2,SM3,SM4)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      CALL COOR (K,I,XI,ETA)
      DDX = X - XI
      DDY = Y - ETA
      RR = DSQRT (DDX*DDX + DDY*DDY)
        IF (RR.LE.0.0) THEN
        SM1 = 0.0
        SM2 = 0.0
```

```
      SM3 = 0.0
      SM4 = 0.0
      ELSE
      SM1 = DWETA (RR,DDY)
      SM2 = DWXI (RR,DDX)
      CALL CALR (K,I,0,0,R,BETA,ALPHA,DX,DY)
      SM3 = DWETAN (RR,DDX,DDY,BETA)
      SM4 = DWXIN (RR,DDX,DDY,BETA)
      END IF
     RETURN
    END
C
C*************************************************************
C
      SUBROUTINE CALR (K,I,L,J,R,BETA,ALPHA,DX,DY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      IF ((K.EQ.L).AND.(I.EQ.J)) GO TO 30
      CALL COOR (K,I,X,Y)
      CALL COOR (L,J,XI,ETA)
      R = DSQRT ((XI-X)**2+(ETA-Y)**2)
      DX = XI-X
      DY = ETA-Y
      IF ((K.EQ.0).OR.(I.EQ.0)) GO TO 10
        X1 = CV(K,1)
        Y1 = CV(K,2)
       IF (K.EQ.KSIDE) THEN
         X2 = CV(1,1)
         Y2 = CV(1,2)
       ELSE
         X2 = CV(K+1,1)
         Y2 = CV(K+1,2)
       END IF
       BETA = ANGLE (X1,Y1,X2,Y2)
   10 IF ((L.EQ.0).OR.(J.EQ.0)) GO TO 20
        X1 = CV(L,1)
        Y1 = CV(L,2)
       IF (L.EQ.KSIDE) THEN
         X2 = CV(1,1)
         Y2 = CV(1,2)
       ELSE
         X2 = CV(L+1,1)
         Y2 = CV(L+1,2)
       END IF
       ALPHA = ANGLE (X1,Y1,X2,Y2)
   20  RETURN
   30  R = 0.0
       DX = 0.0
       DY = 0.0
      RETURN
      END
```

```
C
C*************************************************************
C
      SUBROUTINE CALR1 (K,I,L,J,R,BETA,ALPHA,DX1,DY1,DX2,DY2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      IF ((K.EQ.L).AND.(I.EQ.J)) GO TO 30
      CALL COOR (K,I,X,Y)
      CALL COOR (L,J,XI,ETA)
      R = DSQRT ((XI-X)**2+(ETA-Y)**2)
      CALL COOR1 (L,J,XI1,ETA1,XI2,ETA2)
      DX1 = XI1 - X
      DY1 = ETA1 - Y
      DX2 = XI2 - X
      DY2 = ETA2 - Y
      IF ((K.EQ.0).OR.(I.EQ.0)) GO TO 10
        X1 = CV(K,1)
        Y1 = CV(K,2)
       IF (K.EQ.KSIDE) THEN
         X2 = CV(1,1)
         Y2 = CV(1,2)
       ELSE
         X2 = CV(K+1,1)
         Y2 = CV(K+1,2)
       END IF
       BETA = ANGLE (X1,Y1,X2,Y2)
   10 IF ((L.EQ.0).OR.(J.EQ.0)) GO TO 20
        X1 = CV(L,1)
        Y1 = CV(L,2)
       IF (L.EQ.KSIDE) THEN
         X2 = CV(1,1)
         Y2 = CV(1,2)
       ELSE
         X2 = CV(L+1,1)
         Y2 = CV(L+1,2)
       END IF
       ALPHA = ANGLE (X1,Y1,X2,Y2)
   20  RETURN
   30  R = 0.0
       DX = 0.0
       DY = 0.0
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE COOR (K,I,X,Y)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
```

```
     @CDLJ(50,2),NI(50)
C
      IF (K.EQ.0) GO TO 20
       IF (I.GT.0) GO TO 10
        X = CV(K,1)
        Y = CV(K,2)
        RETURN
   10   X1 = CV(K,1)
        Y1 = CV(K,2)
       IF (K.EQ.KSIDE) THEN
        X2 = CV(1,1)
        Y2 = CV(1,2)
       ELSE
        X2 = CV(K+1,1)
        Y2 = CV(K+1,2)
       END IF
      X = (I-0.5)*(X2-X1)/NELEM(K) + X1
      Y = (I-0.5)*(Y2-Y1)/NELEM(K) + Y1
      RETURN
   20 X = CC(I,1)
      Y = CC(I,2)
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE COOR1 (K,I,XI1,ETA1,XI2,ETA2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      IF (K.EQ.0) GO TO 20
       IF (I.GT.0) GO TO 10
        X = CV(K,1)
        Y = CV(K,2)
        RETURN
   10   X1 = CV(K,1)
        Y1 = CV(K,2)
       IF (K.EQ.KSIDE) THEN
        X2 = CV(1,1)
        Y2 = CV(1,2)
       ELSE
        X2 = CV(K+1,1)
        Y2 = CV(K+1,2)
       END IF
      XI1  = (I-1.0)*(X2-X1)/NELEM(K) + X1
      ETA1 = (I-1.0)*(Y2-Y1)/NELEM(K) + Y1
      XI2  = I*(X2-X1)/NELEM(K) + X1
      ETA2 = I*(Y2-Y1)/NELEM(K) + Y1
      RETURN
   20 X = CC(I,1)
      Y = CC(I,2)
      RETURN
```

```
      END
C
C*************************************************************
C
      SUBROUTINE COORL (I,J,XI1,ETA1,XI2,ETA2,UU)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Q(1)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
C
      ANS  = DBLE(NS(I))
      XI1  = (J-0.5)*(CVL(4,1,I)-CVL(1,1,I))/ANS + CVL(1,1,I)
      ETA1 = (J-0.5)*(CVL(4,2,I)-CVL(1,2,I))/ANS + CVL(1,2,I)
      XI2  = (J-0.5)*(CVL(3,1,I)-CVL(2,1,I))/ANS + CVL(2,1,I)
      ETA2 = (J-0.5)*(CVL(3,2,I)-CVL(2,2,I))/ANS + CVL(2,2,I)
      UU   = (J-0.5)*(U(2,I) - U(1,I))/ANS + U(1,I)
      IF (INCRM(I).EQ.0) GO TO 10
      IF (XI1.LE.XI2) RETURN
      T = XI1
      XI1 = XI2
      XI2 = T
      RETURN
 10   IF (ETA1.LE.ETA2) RETURN
      T = ETA1
      ETA1 = ETA2
      ETA2 = T
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE TRANF (DX,DY,ALPHA,AN,T)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      AN =  DX*DCOS(ALPHA) + DY*DSIN(ALPHA)
      T  = -DX*DSIN(ALPHA) + DY*DCOS(ALPHA)
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE ELEML
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      DO 100 L = 1,KSIDE-1
 100  EL(L) =(DSQRT((CV(L+1,1)-CV(L,1))**2+(CV(L+1,2)-CV(L,2))**2))
     */NELEM(L)
```

```
      L = KSIDE
      EL(L) =(DSQRT((CV(1,1)-CV(L,1))**2+(CV(1,2)-CV(L,2))**2))/NELEM(L)
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE ANOR (L,ALPHA1,ALPHA2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /COORD/ CV(20,2),CC(200,2),EL(20),NDL,CDLI(50,2),
     @CDLJ(50,2),NI(50)
C
      CALL COOR (L,0,X2,Y2)
      IF (L.EQ.1) THEN
      CALL COOR (KSIDE,0,X1,Y1)
      ELSE
      CALL COOR (L-1,0,X1,Y1)
      END IF
      IF (L.EQ.KSIDE) THEN
      CALL COOR (1,0,X3,Y3)
      ELSE
      CALL COOR (L+1,0,X3,Y3)
      END IF
      ALPHA1 = ANGLE (X1,Y1,X2,Y2)
      ALPHA2 = ANGLE (X2,Y2,X3,Y3)
      RETURN
      END
C
C=========================================
C
      SUBROUTINE DISP2 (A,NR,NC)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(NR,1)
      COMMON /LIO/ LR,LW,LC,LM
C
      J1 = 0
      J2 = 0
      ICJ = 7
   10 J1 = J2 + 1
      J2 = J1 + ICJ
      IF (J2.GE.NC) J2 = NC
      WRITE (LM,2000) (J,J=J1,J2)
      WRITE (LM,2001)
      DO 100 I = 1, NR
        IP = I
        WRITE (LM,2010) IP,(A(I,J),J=J1,J2)
  100 CONTINUE
      WRITE (LM,2001)
      WRITE (LM,2001)
      IF (J2.NE.NC) GO TO 10
      RETURN
 2000 FORMAT  (12X,8I12)
```

```
 2001 FORMAT  (/5X,' ')
 2010 FORMAT  (2X,I5,5X,8(1X,E12.5))
      END
C
C=======================================
C
      SUBROUTINE WTAPE2 (A,NR,NC)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(NR,1)
      COMMON /LIO/ LR,LW,LC,LM
C
      J1 = 0
      J2 = 0
      ICJ = 7
   10 J1 = J2 + 1
      J2 = J1 + ICJ
      IF (J2.GE.NC) J2 = NC
C     WRITE (LC,2000) (J,J=J1,J2)
      WRITE (LC,2001)
      DO 100 I = 1, NR
        IP = I
        WRITE (LC,2010) IP,(A(I,J),J=J1,J2)
  100 CONTINUE
      WRITE (LC,2001)
      WRITE (LC,2001)
      IF (J2.NE.NC) GO TO 10
      RETURN
C2000 FORMAT  (12X,8I12)
 2001 FORMAT  (/5X,' ')
 2010 FORMAT  (2X,I5,5X,8(1X,E12.5))
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION ANGLE (X1,Y1,X2,Y2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DX = X2 - X1
      DY = Y2 - Y1
      IF (DX.GE.0.0) GO TO 10
      ANGLE = PI/2.+DATAN(DY/DX)
      RETURN
   10 IF (DX.GT.0.0) GO TO 20
        IF (DY.GE.0.0) THEN
         ANGLE = 0.0
        ELSE
         ANGLE = PI
        END IF
      RETURN
   20 ANGLE = 3.*PI/2.+DATAN(DY/DX)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION FUNCW (R)
```

```
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      FUNCW = R**2*DLOG(R)/(8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWN (R,DX,DY,BETA)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWN = (1.+2.*DLOG(R))*(DX*DCOS(BETA)+DY*DSIN(BETA))/
     @(-8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWXI (R,DX)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWXI = DX*(1. + 2.*DLOG(R))/(8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWX (R,DX)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWX = DX*(1. + 2.*DLOG(R))/(-8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWETA (R,DY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWETA = DY*(1. + 2.*DLOG(R))/(8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWY (R,DY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWY = DY*(1. + 2.*DLOG(R))/(-8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWXIN (R,DX,DY,BETA)
```

```
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWXIN = ((1. + 2.*DLOG(R) + 2.*DX*DX/R/R)*DCOS(BETA) +
     @2.*DX*DY/R/R*DSIN(BETA))/(-8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWETAN (R,DX,DY,BETA)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWETAN = ((1. + 2.*DLOG(R) + 2.*DY*DY/R/R)*DSIN(BETA) +
     @2.*DX*DY/R/R*DCOS(BETA))/(-8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWXIX (R,DX)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWXIX = (1. + 2.*DLOG(R) + 2.*DX*DX/R/R)/(-8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWETAX (R,DX,DY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWETAX = DX*DY/R/R/(-4.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWXIY (R,DX,DY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWXIY = DX*DY/R/R/(-4.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWETAY (R,DY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DWETAY = (1. + 2.*DLOG(R) + 2.*DY*DY/R/R)/(-8.*PI*D)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
      FUNCTION AIN (ALPHA1,ALPHA2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DELTA = ALPHA2 - ALPHA1
      IF (DELTA.GE.0.0) GO TO 10
      DELTA = DELTA + 2.*PI
 10   AIN = PI - DELTA
      IF (AIN.GE.0.0) RETURN
      AIN = AIN + 2.*PI
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION FUNCV (AN,T,K,L)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      IF (K.NE.L) THEN
      FUNCV = (4.*DATAN(T/AN) + 2.*(1.-PR)*AN*T/(T*T+AN*AN))
     @/(-8.*PI)
      ELSE
      FUNCV = 0.0
      END IF
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION FUNCM (AN,T,K,L)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      S = 0.0
      IF (K.EQ.L) GO TO 1
      S = S + 4.*AN*DATAN(T/AN)
 1    S = S - (1.-PR)*T + (1.+PR)*T*DLOG(T*T+AN*AN)
      FUNCM = S/(-8.*PI)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION FUNCR (R,DX,DY,A1,A2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      FUNCR =((DY**2-DX**2)*(DSIN(A1)*DCOS(A1)-DSIN(A2)*DCOS(A2))
     * +2.*DX*DY*((DCOS(A1))**2-(DCOS(A2))**2))*(1.-PR)
     */(4.*PI*R**2)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DMN (AN,T,BETA,ALPHA,K,L)
C
      IMPLICIT REAL*8(A-H,O-Z)
```

```
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      IF (K.NE.L) THEN
      DMN = (4.*DCOS(BETA-ALPHA)*DATAN(T/AN) + (1.+PR)*DSIN(BETA-ALPHA)
     @*DLOG(T*T+AN*AN) - 2.*(1.-PR)*AN*(T*DCOS(BETA-ALPHA)-AN*
     @DSIN(BETA-ALPHA))/(T*T+AN*AN))/(8.*PI)
      ELSE
      DMN = 0.0
      END IF
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DVN (AN,T,BETA,ALPHA)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      RR = AN*AN + T*T
      DVN = ((AN*DSIN(BETA-ALPHA) - T*DCOS(BETA-ALPHA))*
     @(1. + PR + 2.*(1.-PR)*AN*AN/RR)/RR)/(4.*PI)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DRN (R,DX,DY,B,A1,A2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      T = (DSIN(A1)*DCOS(A1)-DSIN(A2)*DCOS(A2))*2.*DX*DY/R/R
     @*(DY*DCOS(B)-DX*DSIN(B))
      T1 = ((DCOS(A1))**2-(DCOS(A2))**2)*(DY*DCOS(B)*(-1.+2.*
     @DX*DX/R/R)+DX*DSIN(B)*(-1.+2.*DY*DY/R/R))
      DRN = (T+T1)*(1.-PR)/(2.*PI*R*R)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DRX (R,DX,DY,A1,A2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DRX = (1. - PR)*((DSIN(A1)*DCOS(A1) - DSIN(A2)*DCOS(A2))*DX
     @*(1. + (DY*DY - DX*DX)/R/R) + ((DCOS(A1))**2 - (DCOS(A2))**2)*
     @DY*(-1. + 2.*DX*DX/R/R))/(2.*PI*R*R)
      RETURN
      END
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DRY (R,DX,DY,A1,A2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DRY = (1. - PR)*((DSIN(A1)*DCOS(A1) - DSIN(A2)*DCOS(A2))*DY
     @*(-1. + (DY*DY - DX*DX)/R/R) + ((DCOS(A1))**2 - (DCOS(A2))**2)*
     @DX*(-1. + 2.*DY*DY/R/R))/(2.*PI*R*R)
```

```
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION ANGLE1 (X1,Y1,X2,Y2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DX = X2 - X1
      DY = Y2 - Y1
C
      IF (DX.GE.0.0) GO TO 10
      ANGLE1 = PI + DATAN(DY/DX)
      RETURN
 10   IF (DX.GT.0.0) GO TO 20
        IF (DY.GE.0.0) THEN
         ANGLE1 = PI/2.0
        ELSE
         ANGLE1 = 3.*PI/2.0
        END IF
      RETURN
 20   ANGLE1 = DATAN(DY/DX)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION AMN (AN,T,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      R = AN*AN + T*T
      T1 = ((DCOS(A))**2 + PR*(DSIN(A))**2)*(-2.*(3. - PR)*T/R + 4.*
     @(1. - PR)*AN*AN*T/R/R) + (1. - PR)*DSIN(2.*A)*(2.*(3. - PR)*AN/R
     @ - 4.*(1. - PR)*AN**3/R/R) + ((DSIN(A))**2 + PR*(DCOS(A))**2)*
     @(2.*(1. + PR)*T/R - 4.*(1. - PR)*AN*AN*T/R/R)
      AMN = D*T1/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION BMN (AN,T,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      R = AN*AN + T*T
      T1 = ((DCOS(A))**2 + PR*(DSIN(A))**2)*(-4.*(1. - 3.*PR)*AN*T/R/R
     @ + 16.*(1. - PR)*AN**3*T/R**3) + (1. - PR)*DSIN(2.*A)*(2.*(1. +
     @PR)/R + 8.*(1. - 2.*PR)*AN*AN/R/R - 16.*(1. - PR)*AN**4/R**3) +
     @((DSIN(A))**2 + PR*(DCOS(A))**2)*(-4.*(1. + PR)*AN*T/R/R - 16.*
     @(1. - PR)*AN**3*T/R**3)
      BMN = D*T1/(8.*PI)
      RETURN
```

```
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION CMN (R,DX,DY,B,A1,A2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      T1 = DSIN(A1)*DCOS(A1) - DSIN(A2)*DCOS(A2)
      T2 = (DCOS(A1))**2 - (DCOS(A2))**2
      CX = T1*(-4. + 12.*DX*DX/R/R + 8.*DX*DX*DY*DY/R**4 - 8.*DX**4/R
     @**4) + 2.*T2*DY*(-6.*DX/R/R + 8.*DX**3/R**4)
      CXY = T1*(8.*DX*DY**3/R**4 - 8.*DX**3*DY/R**4) +
     @ 2.*T2*(-1. + 8.*DX*DX*DY*DY/R**4)
      CY = T1*( 4. - 12.*DY*DY/R/R - 8.*DX*DX*DY*DY/R**4 + 8.*DY**4/R
     @**4) + 2.*T2*DX*(-6.*DY/R/R + 8.*DY**3/R**4)
      CMN = (((DCOS(B))**2 + PR*(DSIN(B))**2)*CX + (1. - PR)*DSIN(2.*B)
     @*CXY + ((DSIN(B))**2 + PR*(DCOS(B))**2)*CY)*D*(1. - PR)/(-4.*PI*
     @R*R)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DDMN (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DXX = 1. + 2.*DLOG(R) + 2.*DX*DX/R/R
      DXY = 2.*DX*DY/R/R
      DYY = 1. + 2.*DLOG(R) + 2.*DY*DY/R/R
      DDMN = (((DCOS(B))**2 + PR*(DSIN(B))**2)*DXX +
     @(1. - PR)*DSIN(2.*B)*DXY + ((DSIN(B))**2 + PR*(DCOS(B))**2)*DYY)
     @/(-8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION EMN (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      EX = -6.*DX/R/R + 4.*DX**3/R**4
      EXY = -2.*DY/R/R + 4.*DX*DX*DY/R**4
      EY = -2.*DX/R/R + 4.*DX*DY*DY/R**4
      EMN = (((DCOS(B))**2 + PR*(DSIN(B))**2)*EX +
     @(1. - PR)*DSIN(2.*B)*EXY + ((DSIN(B))**2 + PR*(DCOS(B))**2)*EY)
     @/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION FMN (R,DX,DY,B)
C
```

```
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      FX = -2.*DY/R/R + 4.*DX*DX*DY/R**4
      FXY = -2.*DX/R/R + 4.*DX*DY*DY/R**4
      FY = -6.*DY/R/R + 4.*DY**3/R**4
      FMN = (((DCOS(B))**2 + PR*(DSIN(B))**2)*FX +
     @(1. - PR)*DSIN(2.*B)*FXY + ((DSIN(B))**2 + PR*(DCOS(B))**2)*FY)
     @/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION AMNT (AN,T,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      R = AN*AN + T*T
      T1 = (-DSIN(2.*A))/2.*(-2.*(3. - PR)*T/R + 4.*
     @(1. - PR)*AN*AN*T/R/R) + DCOS(2.*A)*(2.*(3. - PR)*AN/R
     @ - 4.*(1. - PR)*AN**3/R/R) + (DSIN(2.*A))/2.*
     @(2.*(1. + PR)*T/R - 4.*(1. - PR)*AN*AN*T/R/R)
      AMNT = -D*(1. - PR)*T1/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION BMNT (AN,T,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      R = AN*AN + T*T
      T1 = (-DSIN(2.*A))/2.*(-4.*(1. - 3.*PR)*AN*T/R/R
     @ + 16.*(1. - PR)*AN**3*T/R**3) + DCOS(2.*A)*(2.*(1. +
     @PR)/R + 8.*(1. - 2.*PR)*AN*AN/R/R - 16.*(1. - PR)*AN**4/R**3) +
     @(DSIN(2.*A))/2.*(-4.*(1. + PR)*AN*T/R/R - 16.*
     @(1. - PR)*AN**3*T/R**3)
      BMNT = -D*(1. - PR)*T1/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION CMNT (R,DX,DY,B,A1,A2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      T1 = DSIN(A1)*DCOS(A1) - DSIN(A2)*DCOS(A2)
      T2 = (DCOS(A1))**2 - (DCOS(A2))**2
      CX = T1*(-4. + 12.*DX*DX/R/R + 8.*DX*DX*DY*DY/R**4 - 8.*DX**4/R
     @**4) + 2.*T2*DY*(-6.*DX/R/R + 8.*DX**3/R**4)
      CXY = T1*(8.*DX*DY**3/R**4 - 8.*DX**3*DY/R**4) +
     @ 2.*T2*(-1. + 8.*DX*DX*DY*DY/R**4)
```

```
      CY = T1*( 4. - 12.*DY*DY/R/R - 8.*DX*DX*DY*DY/R**4 + 8.*DY**4/R
     @**4) + 2.*T2*DX*(-6.*DY/R/R + 8.*DY**3/R**4)
      CMNT = (-DSIN(2.*B)*CX/2. + DCOS(2.*B)*CXY + DSIN(2.*B)*CY/2.)
     @*D*(1. - PR)*(1. - PR)/(4.*PI*R*R)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DDMNT (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DXX = 1. + 2.*DLOG(R) + 2.*DX*DX/R/R
      DXY = 2.*DX*DY/R/R
      DYY = 1. + 2.*DLOG(R) + 2.*DY*DY/R/R
      DDMNT = (-DSIN(2.*B)*DXX/2. + DCOS(2.*B)*DXY + DSIN(2.*B)*DYY/2.)
     @*(1. - PR)/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION EMNT (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      EX = -6.*DX/R/R + 4.*DX**3/R**4
      EXY = -2.*DY/R/R + 4.*DX*DX*DY/R**4
      EY = -2.*DX/R/R + 4.*DX*DY*DY/R**4
      EMNT = (-DSIN(2.*B)*EX/2. + DCOS(2.*B)*EXY + DSIN(2.*B)*EY/2.)
     @*(1. - PR)/(-8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION FMNT (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      FX = -2.*DY/R/R + 4.*DX*DX*DY/R**4
      FXY = -2.*DX/R/R + 4.*DX*DY*DY/R**4
      FY = -6.*DY/R/R + 4.*DY**3/R**4
      FMNT = (-DSIN(2.*B)*FX/2. + DCOS(2.*B)*FXY + DSIN(2.*B)*FY/2.)
     @*(1. - PR)/(-8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION AQN (AN,T,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
```

```
      R = AN*AN + T*T
      T1 = ((DCOS(A))**3 + DSIN(A)*DSIN(2.*A)/2.)*8.*(1. - PR)*AN*T/R/R
     @ + ((DSIN(A))**3 + DCOS(A)*DSIN(2.*A)/2.)*(4.*(1. - PR)/R - 8.*
     @(1. - PR)*AN*AN/R/R)
      AQN = D*T1/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION BQN (AN,T,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      R = AN*AN + T*T
      T1 = ((DCOS(A))**3 + DSIN(A)*DSIN(2.*A)/2.)*(-8.*(1. - PR)*T/R/R
     @ + 32.*(1. - PR)*AN*AN*T/R**3) + ((DSIN(A))**3 + DCOS(A)*DSIN(2.*
     @A)/2.)*(24.*(1. - PR)*AN/R/R - 32.*(1. - PR)*AN**3/R**3)
      BQN = D*T1/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION CQN (R,DX,DY,B,A1,A2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      T1 = DSIN(A1)*DCOS(A1) - DSIN(A2)*DCOS(A2)
      T2 = (DCOS(A1))**2 - (DCOS(A2))**2
      CXXX = T1*DX*48.*(-1. + 3.*DX*DX/R/R - 2.*DX**4/R**4) + 2.*
     @T2*DY*6.*(1. - 8.*DX*DX/R/R + 8.*DX**4/R**4)
      CXXY = T1*DY*8.*(-1. + 10.*DX*DX/R/R - 12.*DX**4/R**4) + 2.
     @*T2*DX*2.*(-1. - 8.*DY*DY/R/R + 24.*DX*DX*DY*DY/R**4)
      CXYY = T1*DX*8.*( 1. - 10.*DY*DY/R/R + 12.*DY**4/R**4) + 2.
     @*T2*DY*2.*(-1. - 8.*DX*DX/R/R + 24.*DX*DX*DY*DY/R**4)
      CYYY = T1*DY*48.*( 1. - 3.*DY*DY/R/R + 2.*DY**4/R**4) + 2.*
     @T2*DX*6.*(1. - 8.*DY*DY/R/R + 8.*DY**4/R**4)
      CQN = (((DCOS(B))**3 + DSIN(B)*DSIN(2.*B)/2.)*(CXXX + CXYY) +
     @ ((DSIN(B))**3 + DCOS(B)*DSIN(2.*B)/2.)*(CXXY + CYYY))
     @*D*(1. - PR)/(-4.*PI*R**4)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DDQN (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DXXX = DX*(-6. + 4.*DX*DX/R/R)
      DXXY = DY*(-2. + 4.*DX*DX/R/R)
      DXYY = DX*(-2. + 4.*DY*DY/R/R)
      DYYY = DY*(-6. + 4.*DY*DY/R/R)
      DDQN = (((DCOS(B))**3 + DSIN(B)*DSIN(2.*B)/2.)*(DXXX + DXYY) +
```

```
     @ ((DSIN(B))**3 + DCOS(B)*DSIN(2.*B)/2.)*(DXXY + DYYY))
     @/(-8.*PI*R*R)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION EQN (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      EXXX = 6. - 24.*DX*DX/R/R + 16.*DX**4/R**4
      EXXY = DX*DY*(-12./R/R + 16.*DX*DX/R**4)
      EXYY = -2. + 16.*DX*DX*DY*DY/R**4
      EYYY = DX*DY*(-12./R/R + 16.*DY*DY/R**4)
      EQN = (((DCOS(B))**3 + DSIN(B)*DSIN(2.*B)/2.)*(EXXX + EXYY) +
     @ ((DSIN(B))**3 + DCOS(B)*DSIN(2.*B)/2.)*(EXXY + EYYY))
     @/(8.*PI*R*R)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION FQN (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      FYYY = 6. - 24.*DY*DY/R/R + 16.*DY**4/R**4
      FXXX = DX*DY*(-12./R/R + 16.*DX*DX/R**4)
      FXXY = -2. + 16.*DX*DX*DY*DY/R**4
      FXYY = DX*DY*(-12./R/R + 16.*DY*DY/R**4)
      FQN = (((DCOS(B))**3 + DSIN(B)*DSIN(2.*B)/2.)*(FXXX + FXYY) +
     @ ((DSIN(B))**3 + DCOS(B)*DSIN(2.*B)/2.)*(FXXY + FYYY))
     @/(8.*PI*R*R)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION AVN (AN,T,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      R = AN*AN + T*T
      T1 = ((DCOS(A))**3 + (2.-PR)*DSIN(A)*DSIN(2.*A)/2.)*
     @(4.*(5.-3.*PR)*AN*T/R/R - 16.*(1.-PR)*AN**3*T/R**3)
      T1 = T1 + ((2.-PR)*(DSIN(A))**3 - (0.5-PR)*DCOS(A)*
     @DSIN(2.*A))*(2.*(3.-PR)/R - 8.*(3.-2.*PR)*AN*AN/R/R
     @+ 16.*(1.-PR)*AN**4/R**3)
      T1 = T1 + ((2.-PR)*(DCOS(A))**3 - (0.5-PR)*DSIN(A)*
     @DSIN(2.*A))*(-4.*(3.-PR)*AN*T/R/R + 16.*(1.-PR)*
     @AN**3*T/R**3)
      T1 = T1 + ((DSIN(A))**3 + (2.-PR)*DCOS(A)*DSIN(2.*A)/2.)*
     @(-2.*(1.+PR)/R + 8.*(2.-PR)*AN*AN/R/R - 16.*(1.-PR)*AN**4
     @/R**3)
```

```
      AVN = D*T1/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION BVN (AN,T,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      R = AN*AN + T*T
      T1 = ((DCOS(A))**3 + (2.-PR)*DSIN(A)*DSIN(2.*A)/2.)*
     @(-4.*(1.-3.*PR)*T/R/R + 32.*(2.-3.*PR)*AN*AN*T/R**3
     @- 96.*(1.-PR)*AN**4*T/R**4)
      T1 = T1 + ((2.-PR)*(DSIN(A))**3 - (0.5-PR)*DCOS(A)*
     @DSIN(2.*A))*(12.*(1.-3.*PR)*AN/R/R - 32.*(3.-4.*PR)
     @*AN**3/R**3 + 96.*(1.-PR)*AN**5/R**4)
      T1 = T1 + ((2.-PR)*(DCOS(A))**3 - (0.5-PR)*DSIN(A)*
     @DSIN(2.*A))*(-4.*(1.+PR)*T/R/R - 32.*(1.-2.*PR)*
     @AN*AN*T/R**3 + 96.*(1.-PR)*AN**4*T/R**4)
      T1 = T1 + ((DSIN(A))**3 + (2.-PR)*DCOS(A)*DSIN(2.*A)/2.)*
     @(12.*(1.+PR)*AN/R/R + 32.*(2.-3.*PR)*AN**3/R**3
     @- 96.*(1.-PR)*AN**5/R**4)
      BVN = D*T1/(8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION CVN (R,DX,DY,B,A1,A2)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      T1 = DSIN(A1)*DCOS(A1) - DSIN(A2)*DCOS(A2)
      T2 = (DCOS(A1))**2 - (DCOS(A2))**2
      CXXX = T1*DX*48.*(-1. + 3.*DX*DX/R/R - 2.*DX**4/R**4) + 2.*
     @T2*DY*6.*(1. - 8.*DX*DX/R/R + 8.*DX**4/R**4)
      CXXY = T1*DY*8.*(-1. + 10.*DX*DX/R/R - 12.*DX**4/R**4) + 2.
     @*T2*DX*2.*(-1. - 8.*DY*DY/R/R + 24.*DX*DX*DY*DY/R**4)
      CXYY = T1*DX*8.*( 1. - 10.*DY*DY/R/R + 12.*DY**4/R**4) + 2.
     @*T2*DY*2.*(-1. - 8.*DX*DX/R/R + 24.*DX*DX*DY*DY/R**4)
      CYYY = T1*DY*48.*( 1. - 3.*DY*DY/R/R + 2.*DY**4/R**4) + 2.*
     @T2*DX*6.*(1. - 8.*DY*DY/R/R + 8.*DY**4/R**4)
      CVN = (((DCOS(B))**3 + (2.-PR)*DSIN(B)*DSIN(2.*B)/2.)*CXXX
     @+ ((2.-PR)*(DSIN(B))**3 - (0.5-PR)*DCOS(B)*DSIN(2.*B))*CXXY
     @+ ((2.-PR)*(DCOS(B))**3 - (0.5-PR)*DSIN(B)*DSIN(2.*B))*CXYY
     @+ ((DSIN(B))**3 + (2.-PR)*DCOS(B)*DSIN(2.*B)/2.)*CYYY)
     @*D*(1.-PR)/(-4.*PI*R**4)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DDVN (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
```

```
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      DXXX = DX*(-6. + 4.*DX*DX/R/R)
      DXXY = DY*(-2. + 4.*DX*DX/R/R)
      DXYY = DX*(-2. + 4.*DY*DY/R/R)
      DYYY = DY*(-6. + 4.*DY*DY/R/R)
      DDVN = (((DCOS(B))**3 + (2.-PR)*DSIN(B)*DSIN(2.*B)/2.)*DXXX
     @+ ((2.-PR)*(DSIN(B))**3 - (0.5-PR)*DCOS(B)*DSIN(2.*B))*DXXY
     @+ ((2.-PR)*(DCOS(B))**3 - (0.5-PR)*DSIN(B)*DSIN(2.*B))*DXYY
     @+ ((DSIN(B))**3 + (2.-PR)*DCOS(B)*DSIN(2.*B)/2.)*DYYY)
     @/(-8.*PI*R*R)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION EVN (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      EXXX = 6. - 24.*DX*DX/R/R + 16.*DX**4/R**4
      EXXY = DX*DY*(-12./R/R + 16.*DX*DX/R**4)
      EXYY = -2. + 16.*DX*DX*DY*DY/R**4
      EYYY = DX*DY*(-12./R/R + 16.*DY*DY/R**4)
      EVN = (((DCOS(B))**3 + (2.-PR)*DSIN(B)*DSIN(2.*B)/2.)*EXXX
     @+ ((2.-PR)*(DSIN(B))**3 - (0.5-PR)*DCOS(B)*DSIN(2.*B))*EXXY
     @+ ((2.-PR)*(DCOS(B))**3 - (0.5-PR)*DSIN(B)*DSIN(2.*B))*EXYY
     @+ ((DSIN(B))**3 + (2.-PR)*DCOS(B)*DSIN(2.*B)/2.)*EYYY)
     @/(8.*PI*R*R)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION FVN (R,DX,DY,B)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
C
      FYYY = 6. - 24.*DY*DY/R/R + 16.*DY**4/R**4
      FXXX = DX*DY*(-12./R/R + 16.*DX*DX/R**4)
      FXXY = -2. + 16.*DX*DX*DY*DY/R**4
      FXYY = DX*DY*(-12./R/R + 16.*DY*DY/R**4)
      FVN = (((DCOS(B))**3 + (2.-PR)*DSIN(B)*DSIN(2.*B)/2.)*FXXX
     @+ ((2.-PR)*(DSIN(B))**3 - (0.5-PR)*DCOS(B)*DSIN(2.*B))*FXXY
     @+ ((2.-PR)*(DCOS(B))**3 - (0.5-PR)*DSIN(B)*DSIN(2.*B))*FXYY
     @+ ((DSIN(B))**3 + (2.-PR)*DCOS(B)*DSIN(2.*B)/2.)*FYYY)
     @/(8.*PI*R*R)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION WINT (X,Y,XI,ETA,IZ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
```

```
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
C
      DX = XI - X
      DY = ETA - Y
      IF (INCRM(IZ).EQ.0) GO TO 10
      T = 0.0
      IF (DABS(DY).LE.1.E-30) GO TO 5
      T = 4.*DY**3*DATAN(DX/DY)/3.
 5    IF (DABS(DX).LE.1.E-30) THEN
      WINT = (T - 2.*DX*(DX*DX + 6.*DY*DY)/9.)/(16.*PI*D)
      ELSE
      WINT = (T - 2.*DX*(DX*DX + 6.*DY*DY)/9. + DX/3.*(DX*DX + 3.*DY*DY
     @)*DLOG(DX*DX + DY*DY))/(16.*PI*D)
      END IF
      RETURN
 10   T = 0.0
      IF (DABS(DX).LE.1.E-30) GO TO 15
      T = 4.*DX**3*DATAN(DY/DX)/3.
 15   IF (DABS(DY).LE.1.E-30) THEN
      WINT = (T - 2.*DY*(DY*DY + 6.*DX*DX)/9.)/(16.*PI*D)
      ELSE
      WINT = (T - 2.*DY*(DY*DY + 6.*DX*DX)/9. + DY/3.*(DY*DY + 3.*DX*DX
     @)*DLOG(DX*DX + DY*DY))/(16.*PI*D)
      END IF
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWNINT (X,Y,XI,ETA,B,IZ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
C
      DX = XI - X
      DY = ETA - Y
      RR = DX*DX + DY*DY
      IF (INCRM(IZ).EQ.0) GO TO 20
      T  = 0.0
      IF (DABS(DY).LE.1.E-30) GO TO 5
      T  = 2.*DSIN(B)*DY*DY*DATAN(DX/DY)
 5    IF ((DABS(DX).LE.1.E-30).AND.(DABS(DY).LE.1.E-30)) THEN
      DWNINT = 0.0
      ELSE
      DWNINT = (T + DLOG(RR)*(DCOS(B)*RR/2. + DSIN(B)*DX*DY)
     @ - DX*DY*DSIN(B))/(-8.*PI*D)
      END IF
      RETURN
 20   T = 0.0
      IF (DABS(DX).LE.1.E-30) GO TO 15
      T = 2.*DCOS(B)*DX*DX*DATAN(DY/DX)
 15   IF ((DABS(DY).LE.1.E-30).AND.(DABS(DX).LE.1.E-30)) THEN
      DWNINT = 0.0
```

```
      ELSE
      DWNINT = (T + DLOG(RR)*(DSIN(B)*RR/2. + DX*DY*DCOS(B))
     @ - DX*DY*DCOS(B))/(-8.*PI*D)
      END IF
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWXINT (X,Y,XI,ETA,IZ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
C
      DX = XI - X
      DY = ETA - Y
      RR = DX*DX + DY*DY
      IF (INCRM(IZ).EQ.0) GO TO 10
      IF (RR.LE.1.E-30) THEN
      DWXINT = 0.0
      ELSE
      DWXINT = RR*DLOG(RR)/2./(-8.*PI*D)
      END IF
      RETURN
 10   IF (DABS(DX).LE.1.E-30) THEN
      DWXINT = 0.0
      ELSE
      DWXINT = (-DY*(1. - DLOG(RR)) + 2.*DX*DATAN(DY/DX))*DX
     @/(-8.*PI*D)
      END IF
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DWYINT (X,Y,XI,ETA,IZ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
C
      DX = XI - X
      DY = ETA - Y
      RR = DX*DX + DY*DY
      IF (INCRM(IZ).EQ.0) GO TO 10
      IF (DABS(DY).LE.1.E-30) THEN
      DWYINT = 0.0
      ELSE
      DWYINT = (-DX*(1. - DLOG(RR)) + 2.*DY*DATAN(DX/DY))*DY/
     @(-8.*PI*D)
      END IF
      RETURN
 10   IF (RR.LE.1.E-30) THEN
      DWYINT = 0.0
```

```
      ELSE
      DWYINT = RR*DLOG(RR)/2./(-8.*PI*D)
      END IF
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DIMN (X,Y,XI,ETA,B,IZ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
C
      DX = XI - X
      DY = ETA - Y
      RR = DX*DX + DY*DY
      T1 = (DCOS(B))**2 + PR*(DSIN(B))**2
      T2 = (1. - PR)*DSIN(2.*B)
      T3 = (DSIN(B))**2 + PR*(DCOS(B))**2
      IF (INCRM(IZ).EQ.0) GO TO 20
      T = 0.0
      IF (DABS(DY).LE.1.E-30) GO TO 5
      T = 4.*DY*DATAN(DX/DY)*T3 + DY*DLOG(RR)*T2
 5    IF (DABS(DX).LE.1.E-30) GO TO 10
      DIMN = (T + T1*DX*(1. + DLOG(RR)) - T3*DX*(1. - DLOG(RR)))/
     @(-8.*PI)
      RETURN
 10   DIMN = T/(-8.*PI)
      RETURN
 20   T = 0.0
      IF (DABS(DX).LE.1.E-30) GO TO 25
      T = 4.*DX*DATAN(DY/DX)*T1 + DX*DLOG(RR)*T2
 25   IF (DABS(DY).LE.1.E-30) GO TO 30
      DIMN = (T - T1*DY*(1. - DLOG(RR)) + T3*DY*(1. + DLOG(RR)))/
     @(-8.*PI)
      RETURN
 30   DIMN = T/(-8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DIMNT (X,Y,XI,ETA,B,IZ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
C
      DX = XI - X
      DY = ETA - Y
      RR = DX*DX + DY*DY
      T1 = -DSIN(2.*B)/2.
      T2 =  DCOS(2.*B)
      T3 = -T1
```

```
      IF (INCRM(IZ).EQ.0) GO TO 20
      T = 0.0
      IF (DABS(DY).LE.1.E-30) GO TO 5
      T = 4.*DY*DATAN(DX/DY)*T3 + DY*DLOG(RR)*T2
 5    IF (DABS(DX).LE.1.E-30) GO TO 10
      DIMNT = (T + T1*DX*(1. + DLOG(RR)) - T3*DX*(1. - DLOG(RR)))/
     @(8.*PI)*(1. - PR)
      RETURN
 10   DIMNT = T/(8.*PI)*(1. - PR)
      RETURN
 20   T = 0.0
      IF (DABS(DX).LE.1.E-30) GO TO 25
      T = 4.*DX*DATAN(DY/DX)*T1 + DX*DLOG(RR)*T2
 25   IF (DABS(DY).LE.1.E-30) GO TO 30
      DIMNT = (T - T1*DY*(1. - DLOG(RR)) + T3*DY*(1. + DLOG(RR)))/
     @(8.*PI)*(1. - PR)
      RETURN
 30   DIMNT = T/(8.*PI)*(1. - PR)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DIQN (X,Y,XI,ETA,B,IZ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
C
      DX = XI - X
      DY = ETA - Y
      RR = DX*DX + DY*DY
      T1 = (DCOS(B))**3 + DSIN(B)*DSIN(2.*B)/2.
      T2 = (DSIN(B))**3 + DCOS(B)*DSIN(2.*B)/2.
      IF (INCRM(IZ).EQ.0) GO TO 20
      T  = 0.0
      IF (DABS(DY).LE.1.E-30) GO TO 5
      T  = -4.*DATAN(DX/DY)*T2
 5    IF (DABS(DX).LE.1.E-30) GO TO 10
      DIQN = (T - 2.*DLOG(RR)*T1)/(-8.*PI)
      RETURN
 10   DIQN = T/(-8.*PI)
      RETURN
 20   T = 0.0
      IF (DABS(DX).LE.1.E-30) GO TO 25
      T = -4.*DATAN(DY/DX)*T1
 25   IF (DABS(DY).LE.1.E-30) GO TO 30
      DIQN = (T - 2.*DLOG(RR)*T2)/(-8.*PI)
      RETURN
 30   DIQN = T/(-8.*PI)
      RETURN
      END
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      FUNCTION DIVN (X,Y,XI,ETA,B,IZ)
```

```
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON /CONST/ KSIDE,NELEM(20),NCOL,PR,PI,D,NTEMP
      COMMON /LOAD/ NPL,XP(50,2),PL(50),NZ,NS(20),U(2,20),
     @CVL(4,2,20),INCRM(20),NCM,CM(50,2),CCM(50,2)
C
      DX = XI - X
      DY = ETA - Y
      RR = DX*DX + DY*DY
      T1 = (DCOS(B))**3 + (2.-PR)*DSIN(B)*DSIN(2.*B)/2.
      T2 = (2.-PR)*(DSIN(B))**3 - (0.5-PR)*DCOS(B)*DSIN(2.*B)
      T3 = (2.-PR)*(DCOS(B))**3 - (0.5-PR)*DSIN(B)*DSIN(2.*B)
      T4 = (DSIN(B))**3 + (2.-PR)*DCOS(B)*DSIN(2.*B)/2.
      IF (INCRM(IZ).EQ.0) GO TO 20
      IF (DABS(DY).LE.1.E-30) GO TO 5
      T = -4.*DATAN(DX/DY)*T4 + (T4-T2)*2.*DX*DY/RR
     @ + (T1-T3)*2.*DY*DY/RR - (T1+T3)*DLOG(RR)
      DIVN = T/(-8.*PI)
      RETURN
 5    IF (DABS(DX).LE.1.E-30) GO TO 10
      T = -(T1+T3)*DLOG(RR)
      DIVN = T/(-8.*PI)
      RETURN
 10   DIVN = 0.0
      RETURN
 20   IF (DABS(DX).LE.1.E-30) GO TO 25
      T = -4.*DATAN(DY/DX)*T1 + (T1-T3)*2.*DX*DY/RR
     @ - (T2+T4)*DLOG(RR) + (T4-T2)*2.*DX*DX/RR
      DIVN = T/(-8.*PI)
      RETURN
 25   IF (DABS(DY).LE.1.E-30) GO TO 30
      T = -(T2+T4)*DLOG(RR)
      DIVN = T/(-8.*PI)
      RETURN
 30   DIVN = 0.0
      RETURN
      END
```

APPENDIX D

DATA INPUT

I HEADING

	variable	entry
(1)	HEAD	Title of problem (80 alphabets)

II MASTER CONTROL PARAMETERS

	variable	entry
(1)	KSIDE	Total number of sides of the plate
(2)	NCOL	Total number of supports inside the plate domain
(3)	MODEX	Program execution mode : EQ. 0 problem solution EQ. 1 data check only

III BOUNDARY COORDINATES

	variable	entry
(1)	N	Vertex number
(2)	CV(N,1)	X -ordinate
	CV(N,2)	Y -ordinate
(3)	NELEM(N)	Number of intervals on side number N

IV INTERIOR SUPPORT DATA

	variable	entry
(1)	N	Support number
(2)	CC(N,1)	X -ordinate
	CC(N,2)	Y -ordinate
(3)	STA(N)	Axial stiffness of support
	STR(N,1)	Rotational stiffness of support about x-axis
	STR(N,2)	Rotational stiffness of support about y-axis

V GEOMETRIC AND MATERIAL PROPERTIES INFORMATION

	variable	entry
(1)	TH	Plate thickness
(2)	E	Modulus of elasticity
(3)	PR	Poisson's ratio

VI LOADING DATA

A. Control parameters

	variable	entry
(1)	NPL	Total number of concentrated loads
(2)	NZ	Total number of zones subjected to distributed load
(3)	NCM	Total number of concentrated moments

B. Concentrated load

	variable	entry
(1)	N	Concentrated load number
(2)	PL(N)	Magnitude of z-direction force
(3)	XP(N,1)	X -ordinate
	XP(N,2)	Y -ordinate

C. Distributed load

	variable	entry
(1)	N	Zone number
(2)	INCRM(N)	Dividing into strip on x or y-direction EQ. 0 x-direction EQ. 1 y-direction
(3)	NS(N)	Number of strips
(4)	U(1,N)	Intensity of distributed load at 1st vertex
	U(2,N)	Intensity of distributed load at 3rd vertex
(5)	CVL(1,1,N)	X -ordinate of 1st vertex
	CVL(1,2,N)	Y -ordinate of 1st vertex
	CVL(2,1,N)	X -ordinate of 2nd vertex
	CVL(2,2,N)	Y -ordinate of 2nd vertex
	CVL(3,1,N)	X -ordinate of 3rd vertex
	CVL(3,2,N)	Y -ordinate of 3rd vertex
	CVL(4,1,N)	X -ordinate of 4th vertex
	CVL(4,2,N)	Y -ordinate of 4th vertex

D. Concentrated moment

	variable	entry
(1)	N	Concentrated moment number
(2)	CM(N,1)	Magnitude of x-axis moment
(3)	CM(N,2)	Magnitude of y-axis moment
(4)	CCM(N,1)	X -ordinate
	CCM(N,2)	Y -ordinate

VII SOLUTION OUTPUT

A. Control parameter

	variable	entry
(1)	NDL	Number of solution line to compute deflection and stress resultants

B. Coordinates

	variable	entry
(1)	N	Solution line number
(2)	CDLI(N,1)	X -ordinate of starting point
	CDLI(N,2)	Y -ordinate of starting point
	CDLJ(N,1)	X -ordinate of ending point
	CDLJ(N,2)	Y -ordinate of ending point
(3)	NI(N)	Number of intervals of solution line

DATA FILE

```
  * * * E X A M P L E  2 * * *
4,9,0
1,-10.0,-10.0,10
2,10.0,-10.0,10
3,10.0,10.0,10
4,-10.0,10.0,10
1,-8.0,-8.0,8.06034E+08,1.71786E+08,1.71786E+08
2,0.0,-8.0,8.06034E+08,1.71786E+08,1.71786E+08
3,8.0,-8.0,8.06034E+08,1.71786E+08,1.71786E+08
4,-8.0,0.0,8.06034E+08,1.71786E+08,1.71786E+08
5,0.0,0.0,8.06034E+08,1.71786E+08,1.71786E+08
6,8.0,0.0,8.06034E+08,1.71786E+08,1.71786E+08
7,-8.0,8.0,8.06034E+08,1.71786E+08,1.71786E+08
8,0.0,8.0,8.06034E+08,1.71786E+08,1.71786E+08
9,8.0,8.0,8.06034E+08,1.71786E+08,1.71786E+08
0.25,2.204E+09,0.15
0,1,0
1,1,40,1520.0,1520.0,-10.0,-10.0,10.0,-10.0,10.0,10.0,-10.0,10.0
4
1,0.0,0.5,9.999,0.5,20
2,0.0,4.5,9.999,4.5,20
3,0.001,0.0,9.999,0.0,20
4,0.001,0.001,9.999,9.999,20
```

OUTPUT

EXAMPLE 2

CONTROL PARAMETERS

NUMBER OF SIDES = 4
NUMBER OF SUPPORTS = 9
SOLUTION MODE = 0
EQ. 0 PROBLEM SOLUTION
EQ. 1 DATA CHECK

BOUNDARY DATA

VERTEX NUMBER	COORDINATES X	COORDINATES Y	NUMBER OF INTERVALS
1	-10.00000	-10.00000	10
2	10.00000	-10.00000	10
3	10.00000	10.00000	10
4	-10.00000	10.00000	10

INTERIOR SUPPORT DATA

SUPPORT NUMBER	COORDINATES X	COORDINATES Y	AXIAL STIFFNESS	X-ROTAT. STIFFNESS	Y-ROTAT. STIFFNESS
1	-8.0000	-8.0000	0.80603E+09	0.17179E+09	0.17179E+09
2	0.0000	-8.0000	0.80603E+09	0.17179E+09	0.17179E+09
3	8.0000	-8.0000	0.80603E+09	0.17179E+09	0.17179E+09
4	-8.0000	0.0000	0.80603E+09	0.17179E+09	0.17179E+09
5	0.0000	0.0000	0.80603E+09	0.17179E+09	0.17179E+09
6	8.0000	0.0000	0.80603E+09	0.17179E+09	0.17179E+09
7	-8.0000	8.0000	0.80603E+09	0.17179E+09	0.17179E+09
8	0.0000	8.0000	0.80603E+09	0.17179E+09	0.17179E+09
9	8.0000	8.0000	0.80603E+09	0.17179E+09	0.17179E+09

GEOMETRIC AND MATERIAL PROPERTIES

PLATE THICKNESS	YOUNG*S MODULUS	POISSON*S RATIO
0.25000	0.22040E+10	0.15000

LOADING DATA

TOTAL NUMBER OF CONCENTRATED LOADS = 0
TOTAL NUMBER OF ZONES SUBJECTED TO DISTRIBUTED LOAD = 1
TOTAL NUMBER OF CONCENTRATED MOMENTS = 0

D I S T R I B U T E D L O A D

ZONE NUMBER	DIVIDING DIRECTION	NUMBER OF STRIPS
1	1	40

ZONE	INTENSITY		COORDINATES							
NUMBER	1ST VERTEX	3RD VERTEX	1ST-X	1ST-Y	2ND-X	2ND-Y	3RD-X	3RD-Y	4TH-X	4TH-Y
1	0.15200E+04	0.15200E+04	-10.0000	-10.0000	10.0000	-10.0000	10.0000	10.0000	-10.0000	10.0000

NUMBER OF SOLUTION LINES = 4

S O L U T I O N O U T P U T

LINE	COORDINATES				NUMBER OF
NUMBER	I-X	I-Y	J-X	J-Y	INTERVALS
1	0.0000	0.5000	9.9990	0.5000	20
2	0.0000	4.5000	9.9990	4.5000	20
3	0.0010	0.0000	9.9990	0.0000	20
4	0.0010	0.0010	9.9990	9.9990	20

NO. OF EQUATIONS = 111
REQUIRED STORAGES = 12938
PLATE RIGIDITY (D) = 2935848.252344

SOLUTION LINE NUMBER 1

	W	Mn	Mnt	Qn	Vn
1	0.97244E-03	-0.17098E+05	0.11218E-06	0.49731E-07	0.34008E-07
2	0.15842E-02	-0.10506E+05	-0.33160E+04	0.15733E+05	0.15527E+05
3	0.29931E-02	-0.43336E+04	-0.25263E+04	0.12125E+05	0.15006E+05
4	0.46858E-02	-0.65877E+03	-0.17396E+04	0.84996E+04	0.11186E+05
5	0.63615E-02	0.17502E+04	-0.11980E+04	0.59837E+04	0.80409E+04
6	0.78293E-02	0.33962E+04	-0.80710E+03	0.41309E+04	0.55962E+04
7	0.89604E-02	0.44932E+04	-0.50312E+03	0.26440E+04	0.36006E+04
8	0.96696E-02	0.51396E+04	-0.24772E+03	0.13520E+04	0.18636E+04
9	0.99067E-02	0.53790E+04	-0.14822E+02	0.14566E+03	0.24828E+03
10	0.96530E-02	0.52213E+04	0.21700E+03	-0.10602E+04	-0.13575E+04
11	0.89208E-02	0.46484E+04	0.46958E+03	-0.23525E+04	-0.30672E+04
12	0.77546E-02	0.36050E+04	0.77056E+03	-0.38479E+04	-0.50207E+04
13	0.62360E-02	0.19689E+04	0.11621E+04	-0.57361E+04	-0.74106E+04
14	0.44940E-02	-0.53294E+03	0.17145E+04	-0.83479E+04	-0.10438E+05
15	0.27305E-02	-0.45836E+04	0.24936E+04	-0.12010E+05	-0.13446E+05
16	0.12866E-02	-0.11672E+05	0.25107E+04	-0.12121E+05	-0.71493E+04
17	0.72320E-03	-0.11456E+05	-0.46191E+04	0.21251E+05	0.11950E+05
18	0.96111E-03	-0.18332E+04	-0.24147E+04	0.10336E+05	0.14860E+05
19	0.13185E-02	-0.53258E+03	-0.14426E+04	0.47003E+04	0.69191E+04
20	0.16609E-02	-0.49506E+02	-0.11650E+04	0.21992E+04	0.32008E+04
21	0.19637E-02	0.22176E+03	-0.87550E+03	0.36718E+03	0.10882E+04

SOLUTION LINE NUMBER 2

	W	Mn	Mnt	Qn	Vn
1	0.95223E-02	-0.22693E+04	-0.23123E-07	0.11485E-06	-0.90240E-07
2	0.96538E-02	-0.20119E+04	0.21741E+03	0.51100E+03	0.58532E+02
3	0.10025E-01	-0.13029E+04	0.37321E+03	0.91620E+03	0.14152E+03
4	0.10572E-01	-0.30250E+03	0.43781E+03	0.11479E+04	0.23958E+03
5	0.11204E-01	0.79751E+03	0.41898E+03	0.11892E+04	0.31497E+03
6	0.11822E-01	0.18271E+04	0.34462E+03	0.10614E+04	0.33105E+03
7	0.12332E-01	0.26617E+04	0.24311E+03	0.80430E+03	0.27229E+03
8	0.12660E-01	0.32227E+04	0.13360E+03	0.46187E+03	0.14575E+03
9	0.12753E-01	0.34696E+04	0.25434E+02	0.76052E+02	-0.26960E+02
10	0.12591E-01	0.33924E+04	-0.77584E+02	-0.31428E+03	-0.21902E+03
11	0.12179E-01	0.30090E+04	-0.17164E+03	-0.67215E+03	-0.40629E+03
12	0.11553E-01	0.23666E+04	-0.24779E+03	-0.96180E+03	-0.57453E+03
13	0.10771E-01	0.15467E+04	-0.28928E+03	-0.11510E+04	-0.72587E+03
14	0.99061E-02	0.66733E+03	-0.27393E+03	-0.12185E+04	-0.88074E+03
15	0.90375E-02	-0.12621E+03	-0.18384E+03	-0.11669E+04	-0.10674E+04
16	0.82358E-02	-0.69028E+03	-0.18262E+02	-0.10345E+04	-0.12938E+04
17	0.75507E-02	-0.92844E+03	0.20439E+03	-0.88907E+03	-0.15099E+04
18	0.70018E-02	-0.82486E+03	0.46119E+03	-0.77970E+03	-0.15696E+04
19	0.65769E-02	-0.45428E+03	0.70796E+03	-0.63501E+03	-0.12475E+04
20	0.62387E-02	-0.64145E+02	0.77552E+03	-0.26736E+03	-0.58631E+03
21	0.59465E-02	-0.95240E+02	0.61095E+03	0.51103E+03	0.49914E+03

SOLUTION LINE NUMBER 3

	W	Mn	Mnt	Qn	Vn
1	0.12559E-03	-0.67749E+05	-0.90609E-03	0.16109E+08	0.22956E+08
2	0.97485E-03	-0.10208E+05	-0.15445E-05	0.31779E+05	0.45240E+05
3	0.25676E-02	-0.39241E+04	-0.57960E-06	0.15324E+05	0.21754E+05
4	0.43755E-02	-0.39427E+03	-0.22462E-06	0.95534E+04	0.13507E+05
5	0.61272E-02	0.19355E+04	-0.28682E-07	0.64351E+04	0.90575E+04
6	0.76458E-02	0.35383E+04	0.98691E-07	0.43516E+04	0.61023E+04
7	0.88096E-02	0.46105E+04	0.18837E-06	0.27548E+04	0.38596E+04
8	0.95372E-02	0.52435E+04	0.25440E-06	0.13985E+04	0.19760E+04
9	0.97803E-02	0.54779E+04	0.30464E-06	0.14390E+03	0.24951E+03
10	0.95209E-02	0.53230E+04	0.34492E-06	-0.11119E+04	-0.14711E+04
11	0.87708E-02	0.47615E+04	0.37975E-06	-0.24756E+04	-0.33424E+04
12	0.75725E-02	0.37417E+04	0.41407E-06	-0.41028E+04	-0.55900E+04
13	0.60038E-02	0.21503E+04	0.45382E-06	-0.62942E+04	-0.86453E+04
14	0.41860E-02	-0.26533E+03	0.51012E-06	-0.97974E+04	-0.13577E+05
15	0.23042E-02	-0.41742E+04	0.60904E-06	-0.17202E+05	-0.24082E+05
16	0.66487E-03	-0.12306E+05	0.87335E-06	-0.45834E+05	-0.64865E+05
17	0.86899E-04	-0.53616E+07	0.43201E-03	-0.15411E+11	-0.21961E+11
18	0.66033E-03	-0.86139E+03	-0.41885E-06	-0.33346E+03	-0.19343E+03
19	0.10971E-02	-0.42147E+03	-0.11783E-05	0.40042E+04	0.58843E+04
20	0.15040E-02	0.83146E+02	-0.74107E-05	0.23216E+04	0.36623E+04
21	0.18575E-02	0.32885E+03	-0.21473E+01	0.76069E+03	0.17936E+04

SOLUTION LINE NUMBER 4

	W	Mn	Mnt	Qn	Vn
1	0.12560E-03	-0.64545E+05	0.69797E+01	0.11391E+08	0.16232E+08
2	0.15883E-02	-0.71800E+04	0.69741E+01	0.22208E+05	0.31937E+05
3	0.41003E-02	-0.14036E+04	0.69583E+01	0.10341E+05	0.15293E+05
4	0.66973E-02	0.13398E+04	0.69385E+01	0.60924E+04	0.94533E+04
5	0.89807E-02	0.26974E+04	0.69214E+01	0.38104E+04	0.63273E+04
6	0.10774E-01	0.32761E+04	0.69100E+01	0.23720E+04	0.42779E+04
7	0.12022E-01	0.34364E+04	0.68976E+01	0.13945E+04	0.27399E+04
8	0.12729E-01	0.34335E+04	0.68653E+01	0.68701E+03	0.14614E+04
9	0.12915E-01	0.34273E+04	0.67892E+01	0.11349E+03	0.31693E+03
10	0.12590E-01	0.34633E+04	0.66549E+01	-0.44417E+03	-0.76824E+03
11	0.11747E-01	0.34700E+04	0.64745E+01	-0.10938E+04	-0.18738E+04
12	0.10371E-01	0.32826E+04	0.62935E+01	-0.19443E+04	-0.31338E+04
13	0.84721E-02	0.26664E+04	0.61920E+01	-0.31554E+04	-0.48110E+04
14	0.61231E-02	0.12871E+04	0.63030E+01	-0.51112E+04	-0.75299E+04
15	0.35181E-02	-0.14814E+04	0.69318E+01	-0.91956E+04	-0.13331E+05
16	0.10808E-02	-0.77287E+04	0.95327E+01	-0.24739E+05	-0.35401E+05
17	0.60960E-04	-0.41279E+07	0.48929E+04	-0.83957E+10	-0.11964E+11
18	0.58250E-03	0.45973E+03	-0.17180E+01	-0.17202E+04	-0.26606E+04
19	0.77517E-03	-0.20359E+03	0.27539E+00	-0.48018E+03	-0.13651E+04
20	0.94661E-03	0.12635E+02	-0.61060E+01	-0.10285E+05	-0.17973E+05
21	0.10741E-02	-0.14295E+08	-0.15777E+07	-0.12924E+13	-0.23908E+13

S U P P O R T D I S P L A C E M E N T S A N D R E A C T I O N S

SUPPORT NO.	DEFLECTION	SLOPE W.R.T. X	SLOPE W.R.T. Y	REACTION	MOMENT ABOUT X-AXIS	MOMENT ABOUT Y-AXIS
1	0.66503E-04	0.15590E-03	0.15640E-03	0.53603E+05	0.26868E+05	0.26781E+05
2	0.90677E-04	-0.17084E-13	0.20320E-03	0.73089E+05	0.34908E+05	-0.29347E-05
3	0.66503E-04	-0.15590E-03	0.15640E-03	0.53603E+05	0.26868E+05	-0.26781E+05
4	0.90691E-04	0.20267E-03	0.11293E-13	0.73100E+05	0.19399E-05	0.34815E+05
5	0.12558E-03	-0.46488E-15	-0.77996E-13	0.10122E+06	-0.13399E-04	-0.79860E-07
6	0.90691E-04	-0.20267E-03	-0.22291E-13	0.73100E+05	-0.38292E-05	-0.34815E+05
7	0.66503E-04	0.15590E-03	-0.15640E-03	0.53603E+05	-0.26868E+05	0.26781E+05
8	0.90677E-04	-0.13152E-12	-0.20320E-03	0.73089E+05	-0.34908E+05	-0.22593E-04
9	0.66503E-04	-0.15590E-03	-0.15640E-03	0.53603E+05	-0.26868E+05	-0.26781E+05

VITA

Mr. Yuthana Laokeaw was born on March 1, 1961 in Bangkok and graduated B.Eng. in Civil Engineering from Chulalongkorn University in 1983.