MEEN 673

Term project

Buoyancy-Driven Flow in a Cavity with Partial Heating

Jicheng Lu

525004048

1. Introduction

Thermal-induced buoyancy force for the fluid motion and transport process generated in an enclosure is getting much attention because of its practical significance in science and technology. All electronic devices generate heat when they are used. Surrounding temperature of the electronic components has a direct impact on the performance of the device. Thus, the temperature field and flow condition around the electronic device needs to be well studied for its optimal performance [1].

In this report, the buoyancy-driven natural convection flow is studied due to its complex coupling effect between the flow velocity and thermal fields. We consider a two-dimensional, steady flow of viscous and incompressible fluid in the cavity where the buoyant flow develops because of the thermal-induced density gradient.

Two cases are investigated in this report. The first one is the benchmark case, where the top and bottom surfaces of the square cavity are assumed to be insulated while the two vertical walls are subjected to different temperatures (Fig. 1.1). The second case considers a square cavity with two heating sources on the bottom surface (Fig. 1.2). The heat sources are assumed to be isothermally heated at a constant temperature T_H, while two side walls are cooled at a lower constant temperature T_C. The bottom surface, except the heated part, and the top surface are considered to be insulated.

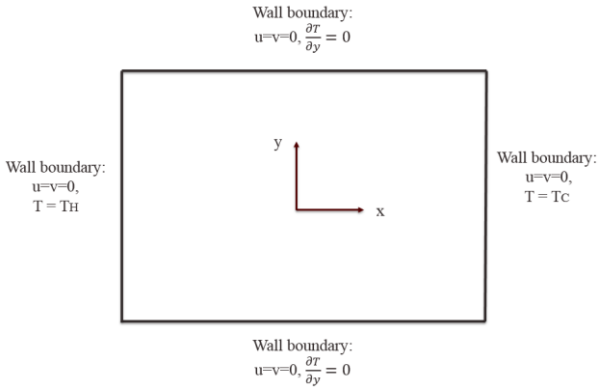


Figure 1.1. Geometry and boundary conditions of the heated cavity (benchmark case).

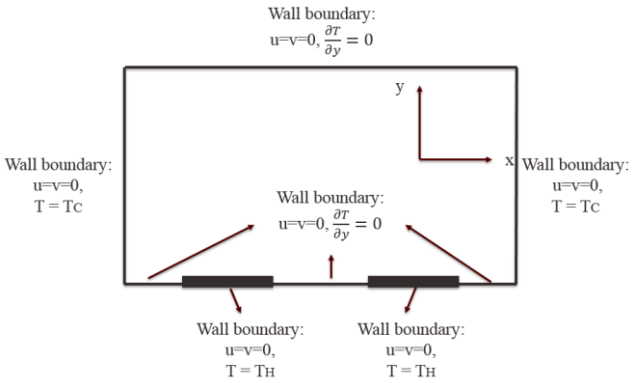


Figure 1.2. Geometry and boundary conditions of the heated cavity (partial heating case).

We investigate the flow and heat transfer characteristics in a square cavity using the non-dimensional Navier-Stokes and energy equations. The penalty finite element model is developed and the Direct Iteration method is applied to solve the problem.

2. Technical description

2.1 Literature review

Some of the most challenging areas in the computational mechanics lie in the analysis of coupled physical phenomena. Typical problems in this area include fluid/structure interaction and coupled thermal/structure analysis. The problems studied in this report is convective heat transfer flow, which involves the coupling of fluid and thermal process.

Depending on the forces for the fluid motion, the convection problems are usually divided into two categories: forced convection and natural convection. Forced convection is due to the pressure or viscous forces on the fluid boundary, which drive the fluid to move forward. Natural convection problems are distinguished by the fluid motion that is generated by the temperature-induced buoyancy forces. In reality, the two types of convection may occur simultaneously, because the two types of driving forces may appear in combinations with varying degrees.

In natural convection area, the most typical problem involves the heat transfer occurring in an enclosure due to the temperature difference across the wall boundaries. In 1970s, Gartling [2] studied both forced and natural convection problems using finite element method. The numerical applications involve the use of temperature-dependent material properties and the inclusion of solid body conduction effects. From that moment on, many researchers began to study the natural convection heat transfer flow in various applications. Davis et al. [3] obtained a benchmark solution to natural convection of air flow in a square cavity with differentially heated walls, where Rayleigh number varied from 10^3 to 10^6. A comparison study was also performed to verify the accuracy of the benchmark solution [4]. Ganzarolli et al. [5] investigated the natural convection of air in an enclosure, which was heated from below and symmetrically cooled from the side walls, using stream function-vorticity formulation. Aydin et al. [6] used the similar numerical technique to study the natural convection problem with heating from one side and cooling from the ceiling. Aydin et al. [7] also studied natural convection of a rectangular enclosure with localized heating from bottom and cooling from the sides.

Due to the complexity of the buoyancy-driven natural convection flow, various numerical techniques were also applied in order to get reasonable results. Nasr et al. [8] used control volume method to solve the full vorticity transport equation with stream function and energy equations, and studied the natural convection in a cavity with heating from lower corner and cooling from ceiling. Barletta et al. [9] used Galerkin finite element method to solve the non-dimensional mass, momentum and energy equations, and investigated the natural convection in a 2D cavity with two vertical isothermal walls and two adiabatic walls. Dalal et al. [10] applied a simple algorithm with deferred quick scheme in curvilinear coordinates and found that the angle of inclination affected the flow and heat transfer rate in the cavity. Snoussi et al. [11] numerically investigated the natural convection flow from the combined buoyancy effect and mass diffusion in a cavity by using the Control-Volume based Finite Element Method (CVFEM) and solve the full vorticity transport equation with stream function, concentration and energy equations.

The objective of this project is to implement the penalty finite element method we have studied in this course and solve the buoyancy-driven flow problems in a cavity with different boundary conditions.

2.2 Governing equations

The governing equations for natural convection flow and conservation of mass, momentum and energy are given as follows:

Continuity equation:
$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0 \tag{1}$$

Momentum equation:
$$\rho\left(u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right)-\frac{\partial \sigma_{xx}}{\partial x}-\frac{\partial \sigma_{xy}}{\partial y}=0$$
$$\rho\left(u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}\right)-\frac{\partial \sigma_{xy}}{\partial x}-\frac{\partial \sigma_{yy}}{\partial y}=\rho g\beta\left(T-T_c\right) \tag{2}$$

Energy equation:
$$\rho c_v\left(u\frac{\partial T}{\partial x}+v\frac{\partial T}{\partial y}\right)=\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right) \tag{3}$$

where the Cauchy stresses are defined by

$$\sigma_{xx}=2\mu\frac{\partial u}{\partial x}-P,\ \ \sigma_{xy}=\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right),\ \ \sigma_{yy}=2\mu\frac{\partial v}{\partial y}-P,$$

$$\tau_{xx}=2\mu\frac{\partial u}{\partial x},\ \ \tau_{xy}=\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right),\ \ \tau_{yy}=2\mu\frac{\partial v}{\partial y}.$$

Note that the Boussinesq approximation is used to include buoyancy force which is caused by temperature dependence of density. The buoyancy force is equal to $\rho g\beta\left(T-T_c\right)$ in the y-direction momentum equation.

Then, we non-dimensionalize the three governing equations using the following dimensionless variables [12]:

$$x^*=\frac{x}{L},\ \ y^*=\frac{y}{L},\ \ u^*=\frac{u\cdot L}{\alpha},\ \ v^*=\frac{v\cdot L}{\alpha},$$
$$P^*=\frac{P\cdot L^2}{\rho\cdot\alpha^2},\ \ T^*=\frac{T-T_C}{T_H-T_C},\ \ \mathrm{Pr}=\frac{\nu}{\alpha},\ \ R_a=\frac{\beta g L^3\left(T_H-T_C\right)}{\alpha\nu}. \tag{4}$$

where x^*, y^*, u^*, v^*, P^*, T^* are the dimensionless coordinates, velocity vectors, pressure and temperature, respectively, and x, y, u, v, P, T denote their physical counterparts. L, ρ are the reference length and reference density, respectively. TC, TH are the reference temperatures. Pr and Ra are the Prandtl number and Rayleigh number, respectively. α, ν, g are the thermal diffusivity, kinematic viscosity and gravitational acceleration.

Applying these dimensionless variables to the governing equations, we obtain the non-dimensional equations:

$$\frac{\partial u^*}{\partial x^*}+\frac{\partial v^*}{\partial y^*}=0 \tag{5}$$

$$u^*\frac{\partial u^*}{\partial x^*}+v^*\frac{\partial u^*}{\partial y^*}=-\frac{\partial P^*}{\partial x^*}+\frac{\partial}{\partial x^*}\left(2\,\mathrm{Pr}\frac{\partial u^*}{\partial x^*}\right)+\frac{\partial}{\partial y^*}\left[\mathrm{Pr}\left(\frac{\partial u^*}{\partial y^*}+\frac{\partial v^*}{\partial x^*}\right)\right] \tag{6}$$

$$u^*\frac{\partial v^*}{\partial x^*}+v^*\frac{\partial v^*}{\partial y^*}=-\frac{\partial P^*}{\partial y^*}+\frac{\partial}{\partial y^*}\left(2\,\mathrm{Pr}\frac{\partial v^*}{\partial y^*}\right)+\frac{\partial}{\partial x^*}\left[\mathrm{Pr}\left(\frac{\partial u^*}{\partial y^*}+\frac{\partial v^*}{\partial x^*}\right)\right]+R_a\,\mathrm{Pr}\cdot T^*$$

$$u^*\frac{\partial T^*}{\partial x^*}+v^*\frac{\partial T^*}{\partial y^*}=\frac{\partial T^{*2}}{\partial x^{*2}}+\frac{\partial T^{*2}}{\partial y^{*2}} \tag{7}$$

2.3. Weak formulations

Before developing the weak formulations, we introduce the penalty function method such that the pressure can be approximated as follows [13]:

$$P^*=-\gamma\left(\frac{\partial u^*}{\partial x^*}+\frac{\partial v^*}{\partial y^*}\right) \tag{8}$$

Substituting Eq. (8) into Eq. (5~7), the governing equations become:

$$u^*\frac{\partial u^*}{\partial x^*}+v^*\frac{\partial u^*}{\partial y^*}=\gamma\frac{\partial}{\partial x^*}\left(\frac{\partial u^*}{\partial x^*}+\frac{\partial v^*}{\partial y^*}\right)+\frac{\partial}{\partial x^*}\left(2\,\mathrm{Pr}\frac{\partial u^*}{\partial x^*}\right)+\frac{\partial}{\partial y^*}\left[\mathrm{Pr}\left(\frac{\partial u^*}{\partial y^*}+\frac{\partial v^*}{\partial x^*}\right)\right] \tag{9}$$

$$u^*\frac{\partial v^*}{\partial x^*}+v^*\frac{\partial v^*}{\partial y^*}=\gamma\frac{\partial}{\partial y^*}\left(\frac{\partial u^*}{\partial x^*}+\frac{\partial v^*}{\partial y^*}\right)+\frac{\partial}{\partial y^*}\left(2\,\mathrm{Pr}\frac{\partial v^*}{\partial y^*}\right)+\frac{\partial}{\partial x^*}\left[\mathrm{Pr}\left(\frac{\partial u^*}{\partial y^*}+\frac{\partial v^*}{\partial x^*}\right)\right]+R_a\,\mathrm{Pr}\cdot T^* \tag{10}$$

$$u^*\frac{\partial T^*}{\partial x^*}+v^*\frac{\partial T^*}{\partial y^*}=\frac{\partial T^{*2}}{\partial x^{*2}}+\frac{\partial T^{*2}}{\partial y^{*2}} \tag{11}$$

Based on the Eq. (9~11), we develop the weak formulations:

$$0=\int_{\Omega^e}\left[w_1\left(u^*\frac{\partial u^*}{\partial x^*}+v^*\frac{\partial u^*}{\partial y^*}\right)+2\,\mathrm{Pr}\frac{\partial w_1}{\partial x^*}\frac{\partial u^*}{\partial x^*}+\mathrm{Pr}\frac{\partial w_1}{\partial y^*}\left(\frac{\partial u^*}{\partial y^*}+\frac{\partial v^*}{\partial x^*}\right)+\gamma\frac{\partial w_1}{\partial x^*}\left(\frac{\partial u^*}{\partial x^*}+\frac{\partial v^*}{\partial y^*}\right)\right]dx^*dy^*-\oint_{\Gamma^e}w_1t_x^*ds \tag{12}$$

$$0=\int_{\Omega^e}\left[w_2\left(u^*\frac{\partial v^*}{\partial x^*}+v^*\frac{\partial v^*}{\partial y^*}\right)+\mathrm{Pr}\frac{\partial w_2}{\partial x^*}\left(\frac{\partial u^*}{\partial y^*}+\frac{\partial v^*}{\partial x^*}\right)+2\,\mathrm{Pr}\frac{\partial w_2}{\partial y^*}\frac{\partial v^*}{\partial y^*}+\gamma\frac{\partial w_2}{\partial y^*}\left(\frac{\partial u^*}{\partial x^*}+\frac{\partial v^*}{\partial y^*}\right)-w_2R_a\,\mathrm{Pr}\cdot T^*\right]dx^*dy^*-\oint_{\Gamma^e}w_2t_y^*ds \tag{13}$$

$$0=\int_{\Omega^e}\left[w_3\left(u^*\frac{\partial T^*}{\partial x^*}+v^*\frac{\partial T^*}{\partial y^*}\right)+\frac{\partial w_3}{\partial x^*}\frac{\partial T^*}{\partial x^*}+k\frac{\partial w_3}{\partial y^*}\frac{\partial T^*}{\partial y^*}\right]dx^*dy^*-\oint_{\Gamma^e}w_3q_n^*ds \tag{14}$$

where

$$t_x^*=\sigma_{xx}^*n_x+\sigma_{xy}^*n_y,\ \ t_y^*=\sigma_{xy}^*n_x+\sigma_{yy}^*n_y,\ \ q_n^*=q_x^*n_x+q_y^*n_y.$$

2.4. Finite element model

The penalty finite element model is obtained by substituting the finite element interpolation for the velocity and temperature field ($u^* = \sum_{j=1}^{n} u_j^* \psi_j, v^* = \sum_{j=1}^{n} v_j^* \psi_j, \mathrm{T}^* = \sum_{j=1}^{n} T_j^* \psi_j$):

$$\begin{bmatrix} K^{11} & K^{12} & K^{13} \\ K^{21} & K^{22} & K^{23} \\ K^{31} & K^{32} & K^{33} \end{bmatrix} \begin{Bmatrix} u^* \\ v^* \\ T^* \end{Bmatrix} = \begin{Bmatrix} F^1 \\ F^2 \\ F^3 \end{Bmatrix} \tag{15}$$

where

$$K_{ij}^{11} = \int_{\Omega^e} \psi_i \left(u^* \frac{\partial \psi_j}{\partial x^*} + v^* \frac{\partial \psi_j}{\partial y^*} \right) dx^* dy^* + \int_{\Omega^e} \mathrm{Pr} \left(2 \frac{\partial \psi_i}{\partial x^*} \frac{\partial \psi_j}{\partial x^*} + \frac{\partial \psi_i}{\partial y^*} \frac{\partial \psi_j}{\partial y^*} \right) dx^* dy^* + \int_{\Omega^e} \gamma \frac{\partial \psi_i}{\partial x^*} \frac{\partial \psi_j}{\partial x^*} dx^* dy^*$$

$$K_{ij}^{12} = \int_{\Omega^e} \mathrm{Pr} \frac{\partial \psi_i}{\partial y^*} \frac{\partial \psi_j}{\partial x^*} dx^* dy^* + \int_{\Omega^e} \gamma \frac{\partial \psi_i}{\partial x^*} \frac{\partial \psi_j}{\partial y^*} dx^* dy^*,$$

$$K_{ij}^{13} = 0, \quad K_{ij}^{21} = K_{ji}^{12},$$

$$K_{ij}^{22} = \int_{\Omega^e} \psi_i \left(u^* \frac{\partial \psi_j}{\partial x^*} + v^* \frac{\partial \psi_j}{\partial y^*} \right) dx^* dy^* + \int_{\Omega^e} \mathrm{Pr} \left(\frac{\partial \psi_i}{\partial x^*} \frac{\partial \psi_j}{\partial x^*} + 2 \frac{\partial \psi_i}{\partial y^*} \frac{\partial \psi_j}{\partial y^*} \right) dx^* dy^* + \int_{\Omega^e} \gamma \frac{\partial \psi_i}{\partial y^*} \frac{\partial \psi_j}{\partial y^*} dx^* dy^*$$

$$K_{ij}^{23} = 0, \quad K_{ij}^{31} = 0, \quad K_{ij}^{32} = 0,$$

$$K_{ij}^{33} = \int_{\Omega^e} \psi_i \left(u^* \frac{\partial \psi_j}{\partial x^*} + v^* \frac{\partial \psi_j}{\partial y^*} \right) dx^* dy^* + \int_{\Omega^e} \left(\frac{\partial \psi_i}{\partial x^*} \frac{\partial \psi_j}{\partial x^*} + \frac{\partial \psi_i}{\partial y^*} \frac{\partial \psi_j}{\partial y^*} \right) dx^* dy^*,$$

$$F_i^1 = \oint_{\Gamma^e} \psi_i t_x^* ds,$$

$$F_i^2 = \int_{\Omega^e} \psi_i R_a \, \mathrm{Pr} \cdot T^* dx^* dy^* + \oint_{\Gamma^e} \psi_i t_y^* ds,$$

$$F_i^3 = \oint_{\Gamma^e} \psi_i q_n^* ds.$$

2.5. Computer implementation

The computer implementation is similar to what we have developed for the fluid mechanics part in the course. The element matrix calculations involve two Gauss loops. All terms except the penalty terms are evaluated by using a full integration loop, while the penalty terms are evaluated using a reduced integration loop. The major difference lies in the fact that all variables we calculated in this program are the non-dimensional variables. Moreover, there is a primary variable (T^*) in the force term F. Box 2.1 gives the main part of the program, and Box 2.2 contains the subroutine to evaluate the coefficient matrix in the penalty finite element model of the Navier-Stokes and energy equations. Note that RAY is the corresponding Rayleigh number, DRAY(I) is increment of the Rayleigh number, and NRAY is the number of the Ra steps. Moreover, PR is the preset Prandtl number.

Box 2.1. FORTRAN statements for the main part of the program.

```
      DO 158 I=1,NEQ
         GPU(I) = 0.0
  158        GCU(I) = 0.0

      RAY = 0.0
      NCOUNT = 0

      DO 175 NL=1,NRAY
      RAY = RAY + DRAY(NL)

      ITER=0
  160 ITER=ITER+1
      NCOUNT = NCOUNT + 1
      IF(ITER.GT.ITMAX)THEN
         WRITE(IT,1200)ITMAX
         WRITE(IT,1050)
         WRITE(IT,940)
         WRITE(IT,670)
         WRITE(IT,940)
         IF(NDF.EQ.1)THEN
         DO I=1,NNM
            WRITE(IT,950)I,(GLXY(I,J),J=1,2),GCU(I)
         ENDDO
         ELSE
         DO I=1,NNM
            I1 = (I-1)*NDF +1
            I2 = I*NDF
            WRITE(IT,950)I,(GLXY(I,J),J=1,2),(GCU(K),K=I1,I2)
         ENDDO
         ENDIF
         WRITE(IT,940)
         STOP
      ENDIF
C
C     Initialize the global coefficient matrices and vectors
C
      DO 180 I=1,NEQ
          DO 180 J=1,NBW
  180        GLK(I,J)=0.0
C
C     Do-loop on the number of ELEMENTS to compute element matrices
C     and their assembly begins here
C
      DO 250 N=1,NEM
      LL = 0
```

```
      DO 200 I=1,NPE
      NI=NOD(N,I)
      IF(NDF.EQ.1)THEN
         ELU(I) = (1.0 - REX)*GCU(NI) + REX*GPU(NI)  ! TRANSFER OF CURRENT SOLUTION
      ELSE
         ND=(NI-1)*NDF
        DO 95 J=1,NDF
            ND=ND+1
           LL=LL+1
   95           ELU(LL)=(1.0 - REX)*GCU(ND) + REX*GPU(ND)
      ENDIF
      ELXY(I,1)=GLXY(NI,1)
      ELXY(I,2)=GLXY(NI,2)
  200 CONTINUE

C
C     Call subroutine ELKMFR (for Rectangular elements) to compute
C     the eLement "K", "M" and "F" matrices.
C

      CALL ELMATRCS2D(ICONV,IEL,NPE,NN,NGPF,NGPR,NONLIN,
     *     NOD,GLXY,NBE,IBN,INOD,BETA,UREF,MAXELM,MAXNOD,MAXCNV,
     *     MAXSPV,MAXSSV,GAMA,RAY,PR)

      IF(NPRNT.EQ.1 .OR. NPRNT.EQ.3) THEN
         IF(N.EQ.1) THEN
C
C     Print element matrices and vectors (only when NPRNT=1 or NPRNT=3)
C
            WRITE(IT,610)
            DO 210 I=1,NN
  210       WRITE(IT,930)(ELK(I,J),J=1,NN)
            WRITE(IT,630)
            WRITE(IT,930)(ELF(I),I=1,NN)
         ENDIF
      ENDIF
C
C     ASSEMBLE element matrices to obtain global matrices:_____________
C
      DO 240 I=1,NPE
         NR=(NOD(N,I)-1)*NDF
         DO 240 II=1,NDF
            NR=NR+1
            L=(I-1)*NDF+II
            GLK(NR,NBW)=GLK(NR,NBW)+ELF(L)
            DO 220 J=1,NPE
```

```
                NCL=(NOD(N,J)-1)*NDF
                DO 220 JJ=1,NDF
                   M=(J-1)*NDF+JJ
                   NC=NCL-NR+JJ+NHBW
                   IF(NC.GT.0)THEN
                      GLK(NR,NC)=GLK(NR,NC)+ELK(L,M)
                  ENDIF
  220        CONTINUE
  240 CONTINUE
  250 CONTINUE

C
C     Print global matrices when NPRNT > 2
C
      IF(NPRNT.GE.2) THEN
         WRITE(IT,640)
         DO 260 I=1,NEQ
  260    WRITE(IT,930) (GLK(I,J),J=1,NBW-1) ! Global coefficient matrix
         WRITE(IT,650)
         WRITE(IT,930) (GLK(I,NBW),I=1,NEQ) ! Global source vector
      ENDIF
C
C     Impose BOUNDARY CONDITIONS on primary and secondary variables
C
      CALL BNDRYUNSYM(NONLIN,MAXNEQ,MAXFBW,MAXSPV,MAXSSV,NDF,NHBW,GLK,
     1                NSPV,NSSV,ISPV,ISSV,VSPV,VSSV)
C
C     Call subroutine SOLVRUNS to solve unsymmetric system of linear equations
C     The solution is returned in the array GLK(I,NBW)
C
C      IRES=0
      CALL SOLVRUNS(GLK,MAXNEQ,MAXFBW,NEQ,NHBW)


C     Save the previous iteration solution and update the current one

      IF(NONLIN.GT.0)THEN
         DO 270 I=1,NEQ
              GPU(I) = GCU(I)
C             GPU(I) = REX*GPU(I) + (1.0 - REX)*GCU(I)
C             !GPU=previous iteration solution; GCU=current solution
             IF(NONLIN.EQ.1)THEN
                GCU(I)=GLK(I,NBW)! GLK(I,NBW) is the full solution
             ELSE
                GCU(I)=GCU(I)+GLK(I,NBW) ! GLK(I,NBW) is the solution increment
             ENDIF
```

```
  270    CONTINUE

         IF(ITER.EQ.1)THEN
           IF(NONLIN.GT.1)THEN ! Setting the specified values to zero after
             DO I=1,NSPV       ! the first iteration of Newton's method
             VSPV(I)=0.0
             END DO
           ENDIF
C            WRITE(IT,1000)
C            WRITE(IT,940)
C            WRITE(IT,670)
C            WRITE(IT,940)
C            IF(NDF.EQ.1)THEN
C            DO I=1,NNM
C               WRITE(IT,950)I,(GLXY(I,J),J=1,2),GCU(I)
C            ENDDO
C            ELSEIF(NDF.EQ.2)THEN
C            DO I=1,NNM
C            I1 = (I-1)*NDF + 1
C            I2 = I*NDF
C               WRITE(IT,950)I,(GLXY(I,J),J=1,2),(GCU(K),K=I1,I2)
C            ENDDO
C            WRITE(IT,940)
         ENDIF

C          PRINT*,GPU(1),GLU(1)
C
C     Test for the convergence of the solution
C
         IF(NONLIN.GT.0)THEN
            DNORM = 0.0
            DINORM = 0.0
            DO 280 I=1,NEQ
               DNORM = DNORM + GCU(I)*GCU(I)
               IF(NONLIN.EQ.1)THEN
                  DINORM = DINORM + (GCU(I)-GPU(I))*(GCU(I)-GPU(I))
               ELSE
                  DINORM = DINORM + GLK(I,NBW)*GLK(I,NBW)
               ENDIF
  280       CONTINUE

            ERROR=DSQRT(DINORM/DNORM)
            WRITE(IT,1100)ITER,ERROR
           IF(ERROR.GT.EPS)THEN
              GOTO 160
            ENDIF
```

```
          ENDIF
         ENDIF
C
C     Print the solution (i.e., nodal values of the primary variables)
C
      IF(NDF.EQ.1)THEN
      WRITE(IT,660)
      WRITE(IT,940)
      WRITE(IT,670)
      WRITE(IT,940)
      DO 300 I=1,NNM
  300 WRITE(IT,950)I,(GLXY(I,J),J=1,2),GCU(I)
      WRITE(IT,940)

      ELSEIF(NDF.EQ.3)THEN
      WRITE(IT,660)
      WRITE(IT,1175)RAY
      WRITE(IT,940)
      WRITE(IT,680)
      WRITE(IT,940)
      DO 301 I=1,NNM
      I1 = (I-1)*NDF+1
      I2 = I*NDF
  301 WRITE(IT,950)I,(GLXY(I,J),J=1,2),(GCU(K),K=I1,I2)
      WRITE(IT,940)
      ENDIF
```

Box 2.1. FORTRAN statements for the calculation of the element matrix of the penalty finite element model.

```
      SUBROUTINE ELMATRCS2D(ICONV,IEL,NPE,NN,NGPF,NGPR,NONLIN,
     *     NOD,GLXY,NBE,IBN,INOD,BETA,UREF,MAXELM,MAXNOD,MAXCNV,
     *     MAXSPV,MAXSSV,GAMA,RAY,PR)
C     _________________________________________________________________
C
C     Element calculations based on linear and quadratic rectangular
C               elements with isoparametric formulation.
C     _________________________________________________________________
C
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/STF/ELF(27),ELK(27,27),ELXY(9,2),ELU(27)
      COMMON/SHP/SFL(9),GDSFL(2,9)
      COMMON/SOURCE/FX0,FY0,FXX,FXY,FYX,FYY
      DIMENSION IBN(MAXCNV),INOD(MAXCNV,3),BETA(MAXCNV),UREF(MAXCNV)
      DIMENSION NOD(MAXELM,9),GLXY(MAXNOD,2)
      DIMENSION GAUSPT(5,5),GAUSWT(5,5),TANG(27,27)
```

```
      COMMON/IO/IN,IT
C
      DATA GAUSPT/5*0.0D0, -0.57735027D0, 0.57735027D0, 3*0.0D0,
     2  -0.77459667D0, 0.0D0, 0.77459667D0, 2*0.0D0, -0.86113631D0,
     3  -0.33998104D0, 0.33998104D0, 0.86113631D0, 0.0D0, -0.90617984D0,
     4  -0.53846931D0,0.0D0,0.53846931D0,0.90617984D0/
C
      DATA GAUSWT/2.0D0, 4*0.0D0, 2*1.0D0, 3*0.0D0, 0.55555555D0,
     2   0.88888888D0, 0.55555555D0, 2*0.0D0, 0.34785485D0,
     3 2*0.65214515D0, 0.34785485D0, 0.0D0, 0.23692688D0,
     4   0.47862867D0, 0.56888888D0, 0.47862867D0, 0.23692688D0/
C
C     Initialize the arrays
C
      NDF = NN/NPE

      DO 100 I = 1,NN
         ELF(I)  = 0.0
      DO 100 J = 1,NN
         IF(NONLIN.GT.1)THEN
            TANG(I,J)=0.0
         ENDIF
  100    ELK(I,J)= 0.0
C
C     Do-loops on numerical (Gauss) integration begin here. Subroutine
C     SHPRCT (SHaPe functions for ReCTangular elements) is called here
C
      DO 200 NI = 1,NGPF
      DO 200 NJ = 1,NGPF
         XI  = GAUSPT(NI,NGPF)
         ETA = GAUSPT(NJ,NGPF)
         CALL INTERPLN2D(NPE,XI,ETA,DET,ELXY)
         CNST = DET*GAUSWT(NI,NGPF)*GAUSWT(NJ,NGPF)
C
C     Compute x, y, U, UX, UY needed
C
C*        ***** YOUR STATEMENTS *****
         X = 0.0
         Y = 0.0
         DO 30 I=1,NPE
         X = X + ELXY(I,1)*SFL(I)
30       Y = Y + ELXY(I,2)*SFL(I)

         FX = FX0+ FXX*X + FXY*Y
         FY = FY0+ FYX*X + FYY*Y
```

```
          IF(NONLIN.GT.0)THEN
          U = 0.0
          V = 0.0
          T = 0.0
          DO 40 I=1,NPE
          L = (I-1)*NDF + 1
          U = U + ELU(L)*SFL(I)
          V = V + ELU(L+1)*SFL(I)
40        T = T + ELU(L+2)*SFL(I)
          ENDIF

C
C     Define the coefficients of the differential equation
C
      II = 1
      DO 90 I=1,NPE
      L = (I-1)*NDF + 1
      JJ=1
      ELF(II) = ELF(II) + FX*SFL(I)*CNST
      ELF(II+1) = ELF(II+1) + FY*SFL(I)*CNST +
     *                        RAY*PR*T*SFL(I)*CNST
      DO 80 J=1,NPE
      S00 = SFL(I)*SFL(J)*CNST
      S11 = GDSFL(1,I)*GDSFL(1,J)*CNST
      S22 = GDSFL(2,I)*GDSFL(2,J)*CNST
      S12 = GDSFL(1,I)*GDSFL(2,J)*CNST
      S21 = GDSFL(2,I)*GDSFL(1,J)*CNST

      ELK(II,JJ) = ELK(II,JJ) + PR*(2.0*S11 + S22)
      ELK(II+1,JJ+1) = ELK(II+1,JJ+1) + PR*(S11 + 2.0*S22)
      ELK(II,JJ+1) = ELK(II,JJ+1) + PR*S21
      ELK(II+1,JJ) = ELK(II+1,JJ) + PR*S12
C      ELK(II+1,JJ+2) = ELK(II+1,JJ+2) - RHO*TBETA*GA*S00
      ELK(II+2,JJ+2) = ELK(II+2,JJ+2) + (S11 + S22)
      IF(NONLIN.GT.0)THEN
          CNV = SFL(I)*(U*GDSFL(1,J)+V*GDSFL(2,J))*CNST
          ELK(II,JJ) = ELK(II,JJ) + CNV
          ELK(II+1,JJ+1) = ELK(II+1,JJ+1) + CNV
          ELK(II+2,JJ+2) = ELK(II+2,JJ+2) + CNV
      ENDIF
80      JJ = NDF*J + 1
90      II = NDF*I + 1
200   CONTINUE
```

```
C     Evaluate the penalty terms using reduced integration
      DO 150 NI = 1,NGPR
      DO 150 NJ = 1,NGPR
         XI  = GAUSPT(NI,NGPR)
         ETA = GAUSPT(NJ,NGPR)
         CALL INTERPLN2D(NPE,XI,ETA,DET,ELXY)
         CNST = DET*GAUSWT(NI,NGPR)*GAUSWT(NJ,NGPR)

         X = 0.0
         Y = 0.0
         DO 110 I=1,NPE
         X = X + ELXY(I,1)*SFL(I)
110      Y = Y + ELXY(I,2)*SFL(I)

         II = 1
         DO 95 I=1,NPE
         JJ=1
         DO 85 J=1,NPE
         S11 = GDSFL(1,I)*GDSFL(1,J)*CNST
         S22 = GDSFL(2,I)*GDSFL(2,J)*CNST
         S12 = GDSFL(1,I)*GDSFL(2,J)*CNST
         S21 = GDSFL(2,I)*GDSFL(1,J)*CNST

         ELK(II,JJ) = ELK(II,JJ) + GAMA*S11
         ELK(II+1,JJ+1) = ELK(II+1,JJ+1) + GAMA*S22
         ELK(II,JJ+1) = ELK(II,JJ+1) + GAMA*S12
         ELK(II+1,JJ) = ELK(II+1,JJ) + GAMA*S21
85       JJ = NDF*J + 1
95       II = NDF*I + 1
150   CONTINUE
      RETURN
      END
```

3. Results and Discussion

The non-dimensional Navier-Stokes and energy equations (Eq. (5~7)) are solved numerically using Direct Iteration method. The range of Rayleigh number varies from 10^3 to 10^5, based on the cavity height (Pr = 0.71). We use 20x20 nine-node quadratic element mesh for the low Ra (e.g. Ra = 10^3, 10^4) and 50x50 nine-node quadratic element mesh for the high Ra (e.g. Ra = 10^5). The thermal field and flow field are investigated in this section.

3.1 Benchmark case

The benchmark case is first studied, where the top and bottom surfaces of the square cavity are assumed to be insulated while the two vertical walls are subjected to different temperature (Fig. 1.1). Fig. 3.1 presents the isotherms and streamlines for Ra = 10^3. It can be observed that the flow is rotating clockwise inside the cavity and the thermal field is only slightly perturbed from the flow conduction. Fig. 3.2 describes the isotherms and streamlines for Ra = 10^4. Note that there is a stronger flow disturbance than the case where Ra equals 10^3 and the thermal field is more strongly affected by the conduction effect. However, due to the limited time and computing resource, the results of Ra = 10^5 cannot be achieved. The thermal field and flow field with Ra = $6x10^4$ are shown in Fig. 3.3. We can observe that at the higher Rayleigh number, the thermal field nearly becomes vertically stratified along the two vertical wall. Further computation needs to be performed to reach higher Rayleigh number.

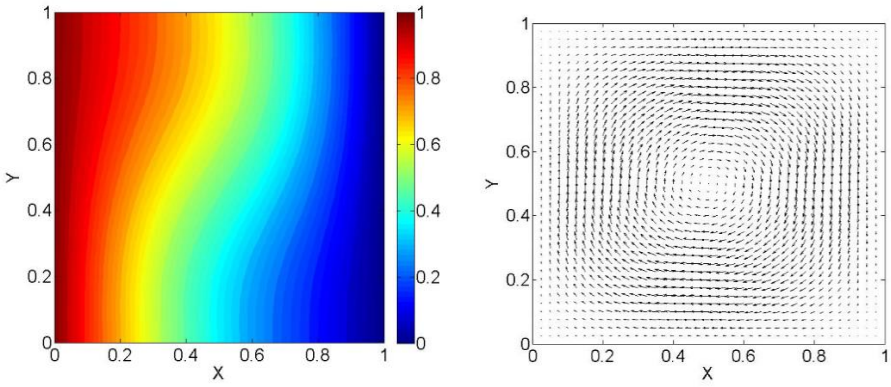


Figure 3.1. Isotherm (left) and velocity vector (right) for Ra=10^3 (Pr = 0.71).

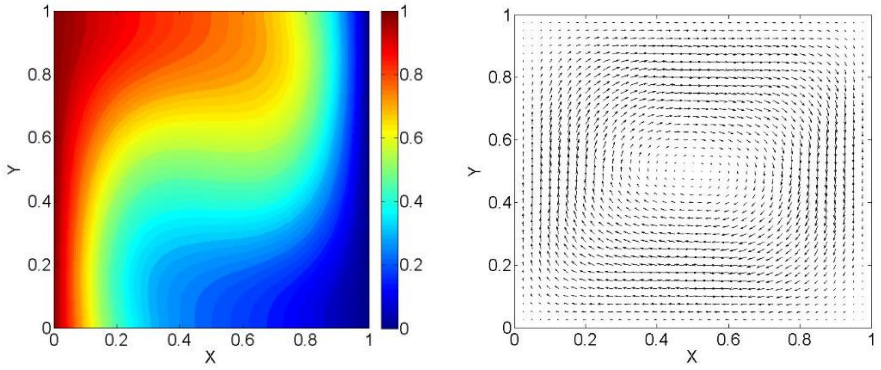


Figure 3.2. Isotherm (left) and velocity vector (right) for Ra=10^4 (Pr = 0.71).

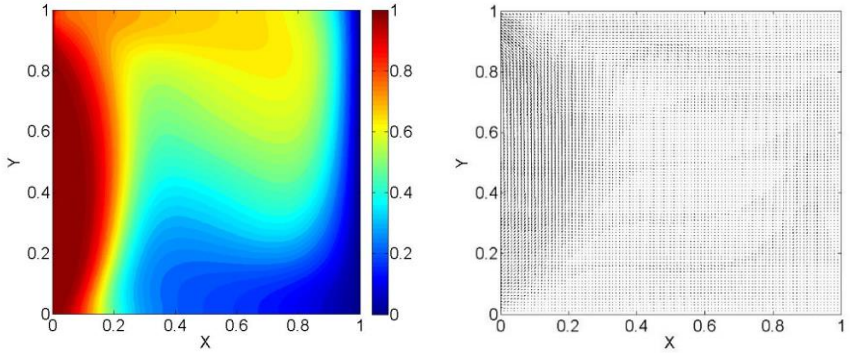
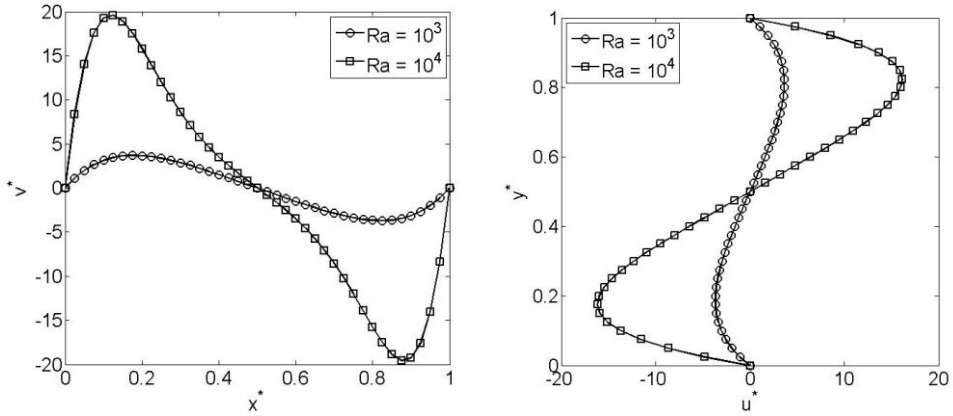


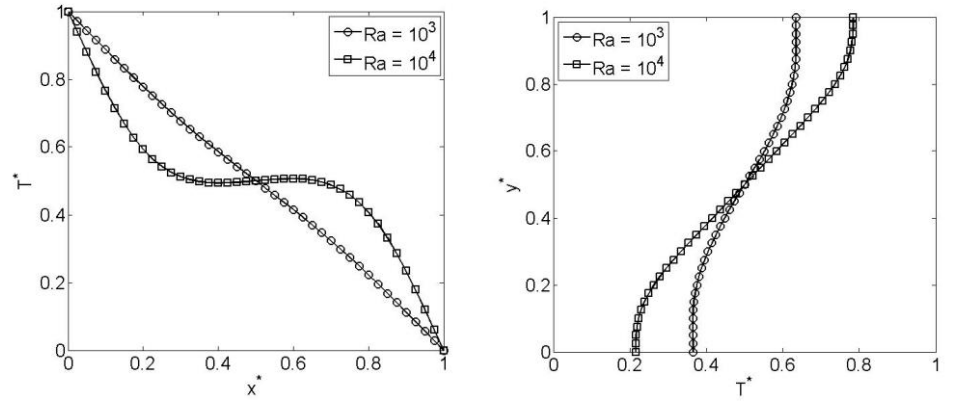
Figure 3.3. Isotherm (left) and velocity vector (right) for Ra=$6x10^4$ (Pr = 0.71).

We compare the numerical results of heat cavity with different Rayleigh numbers (e.g. Ra = 10^3, 10^4). Fig. 3.4(a) presents the distribution of non-dimensional velocity v^* along the line $y^* = 0.5$, while Fig. 3.4(b) shows the distribution of non-dimensional velocity u^* along the line $x^* = 0.5$. It can be observed that the flow field is more perturbed with higher Rayleigh number, especially the region close to the wall boundaries. Fig. 3.5 presents the temperature profile along the line at $y^* = 0.5$ and $x^* = 0.5$ with different Rayleigh number. The temperature distributions clearly indicate the effect of Rayleigh number on the thermal field in the cavity.



(a) (b)

Figure 3.4. Velocity at different locations for the benchmark case: (a) vertical velocity, v^*, along the line y^*=0.5; (b) horizontal velocity, u^*, along the line x^*=0.5. (Ra = 10^3, 10^4, Pr = 0.71)



(a) (b)

Figure 3.5. Temperature at different locations for the benchmark case: (a) temperature, T^*, along the line y^*=0.5; (b) temperature, T^*, along the line x^*=0.5. (Ra = 10^3, 10^4, Pr = 0.71)

3.2 Partial heating case

The partial heating case is studied in this section, where two heating sources are located on the bottom surface (Fig. 1.2). The two heat sources are assumed to be isothermally heated at a higher temperature T_H, while two side walls are cooled at a lower temperature T_C. The bottom surface, except the heated part, and the top surface are insulated. Note that the two heat sources with equal length are symmetric with respect to the middle vertical plane.

Fig. 3.6 presents the isotherms and velocity vectors respectively for Ra = 10^3. In this case, the flow is dominated by the conduction effect. Note that the streamlines form two symmetric vortices: the left one is rotating in counter-clockwise direction (primary vortex) and the right one is rotating in clockwise direction (secondary vortex). In Fig. 3.7, we obtain similar patterns of the velocity field while the thermal field is slightly perturbed due to the increasing Rayleigh number. However, when Ra reaches 10^5, the thermal field is strongly affected by the flow inside the enclosure, while there still two vortices rotating in the cavity. This result is different with [1], where the secondary vortex starts to decentralize and the corresponding thermal field shows a counter-clockwise flow of the fluid. Thus, further investigation is needed, including increasing the grid number or using higher-order finite elements.

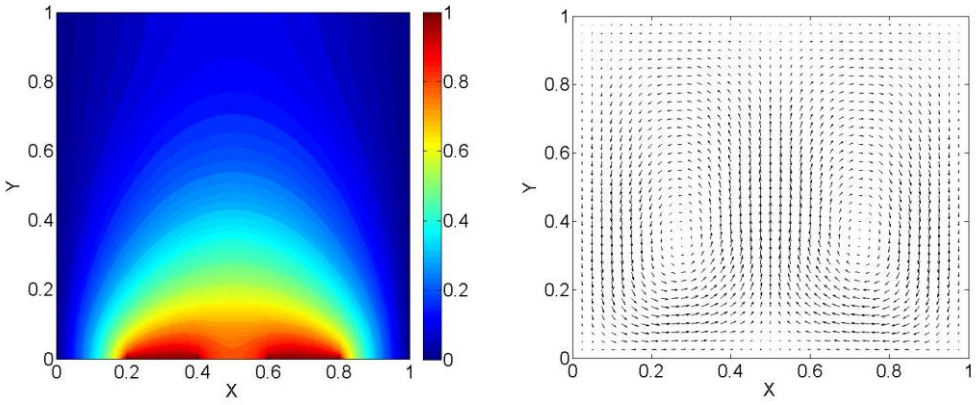


Figure 3.6. Isotherm (left) and velocity vector (right) for Ra=10^3 (Pr = 0.71).

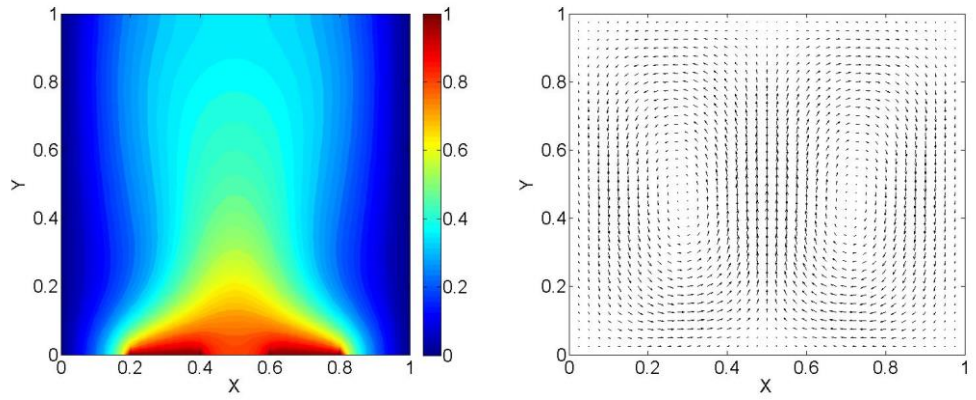


Figure 3.7. Isotherm (left) and velocity vector (right) for Ra=10^4 (Pr = 0.71).

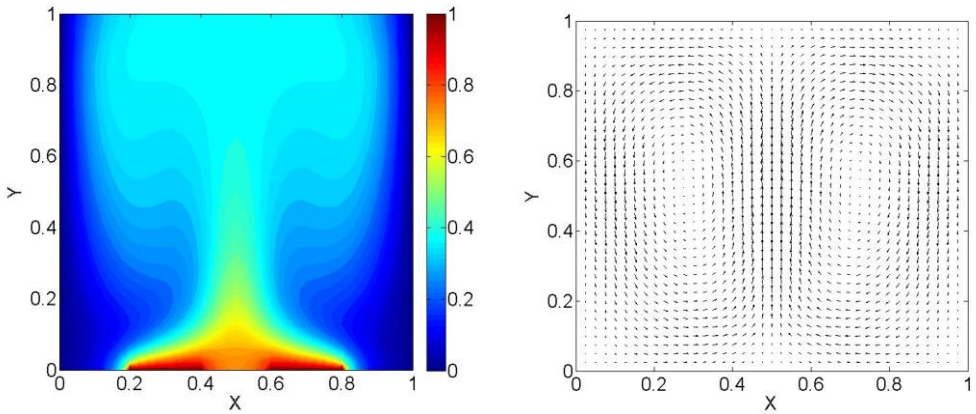
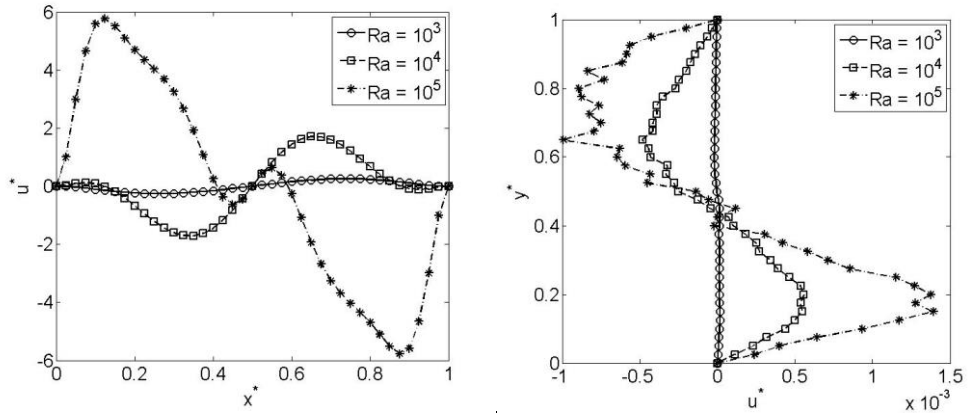


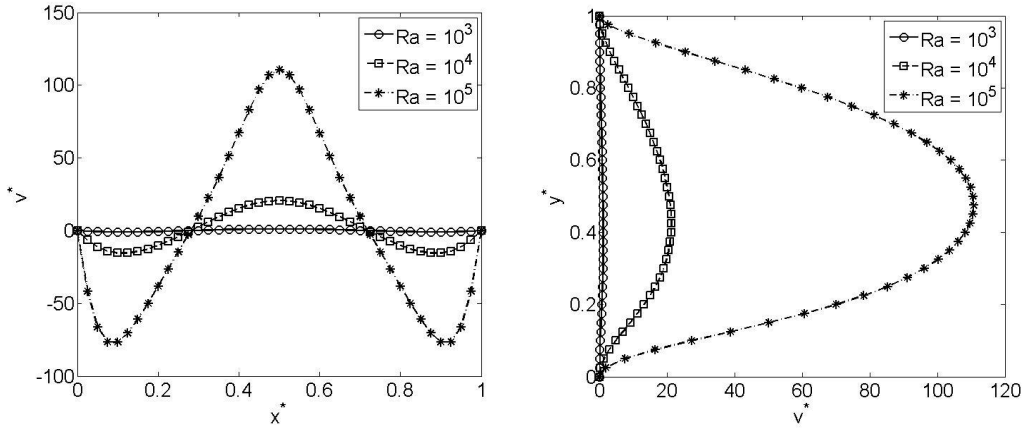
Figure 3.8. Isotherm (left) and velocity vector (right) for Ra=1x10^5 (Pr = 0.71).

Fig. 3.9 illustrates the horizontal velocity profiles at two symmetry surfaces: y^*=0.5 and x^*=0.5, while Fig. 3.10 illustrates the vertical velocity profiles at the symmetry surfaces. We can observe that the distribution of the velocity is a sinusoid-like curve. As the Rayleigh number increases, the maximum absolute values of velocity are increasing as well, which indicates the effect of the buoyancy force on the flow field inside the cavity. Moreover, Fig. 3.11 shows the temperature profiles at the symmetry surfaces. The numerical results clearly indicate the effect of buoyancy force on the thermal field. The temperature profiles at the symmetry surfaces start to strongly oscillate as the Rayleigh number increase.



(a) (b)

Figure 3.9. Horizontal velocity at different locations for the partial heating case: (a) horizontal velocity, u^*, along the line y^*=0.5; (b) horizontal velocity, u^*, along the line x^*=0.5. (Ra = 10^3, 10^4, and 10^5, Pr = 0.71)



(a) (b)

Figure 3.10. Vertical velocity at different locations for the partial heating case: (a) vertical velocity, v^*, along the line y^*=0.5; (b) vertical velocity, v^*, along the line x^*=0.5. (Ra = 10^3, 10^4, and 10^5, Pr = 0.71)

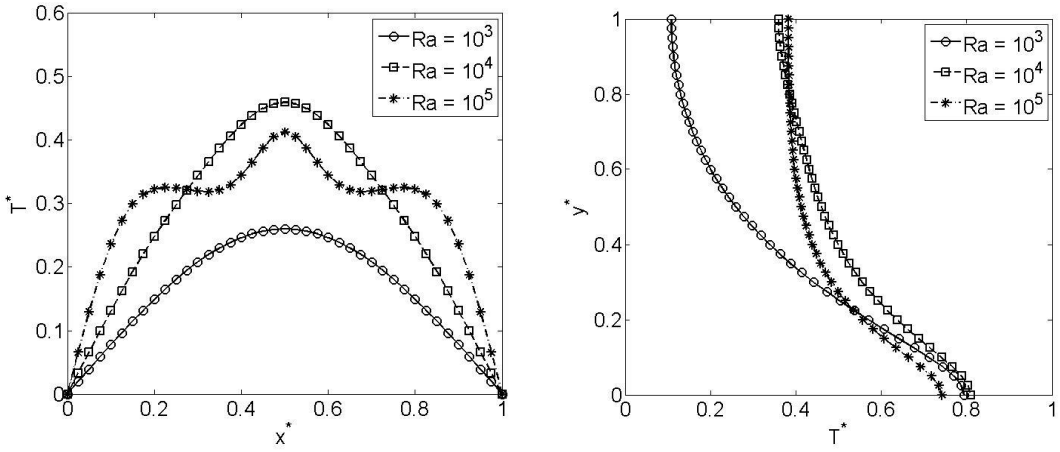


Figure 3.11. Temperature at different locations for the partial heating case: (a) temperature, T^*, along the line $y^*=0.5$; (b) temperature, T^*, along the line $x^*=0.5$. ($Ra = 10^3$, 10^4, and 10^5, $Pr = 0.71$)

6. Conclusions

In this report, we study the buoyancy-driven natural convection flow. A two-dimensional, steady flow of viscous and incompressible fluid in the cavity is considered where the buoyant flow develops because of the thermal-induced density gradient.

Two cases are investigated in this report. The first one is the benchmark case, where the top and bottom surfaces of the square cavity are assumed to be insulated while the two vertical walls are subjected to different temperatures. We observed that flow disturbance becomes stronger as the Rayleigh number increases. For higher Rayleigh number (e.g. $Ra = 10^5$), the actual results cannot be achieved due to the limited time and computing resource. Thus, we just present the results ($Ra = 6x10^4$) as far as I can. Further study is required to get the practical results.

The second case is a square cavity with two heating sources on the bottom surface while the two vertical walls are heated at lower temperature. For lower Rayleigh number, two symmetric vortices are rotating in the opposite direction inside the cavity, while for the higher Rayleigh number, both the flow field and thermal field are strongly affected by the buoyancy force. Moreover, the results for the higher Rayleigh number need further study and verification.

Reference

[1] Zaman, F.S., Turja, T.S. and Molla, M.M., 2013. Buoyancy driven natural convection flow in an enclosure with two discrete heating from below. *Procedia Engineering*, *56*, pp.104-111.

[2] Gartling, D.K., 1977. Convective heat transfer analysis by the finite element method. *Computer Methods in Applied Mechanics and Engineering*, *12*(3), pp.365-382.

[3] de Vahl Davis, G., 1983. Natural convection of air in a square cavity: a bench mark numerical solution. *International Journal for numerical methods in fluids*, *3*(3), pp.249-264.

[4] de Vahl Davis, G. and Jones, I.P., 1983. Natural convection in a square cavity: a comparison exercise. *International Journal for numerical methods in fluids*, *3*(3), pp.227-248.

[5] Ganzarolli, M.M. and Milanez, L.F., 1995. Natural convection in rectangular enclosures heated from below and symmetrically cooled from the sides. *International Journal of Heat and Mass Transfer*, *38*(6), pp.1063-1073.
[6] Aydin, O., Ünal, A. and Ayhan, T., 1999. Natural convection in rectangular enclosures heated from one side and cooled from the ceiling. *International Journal of Heat and Mass Transfer*, *42*(13), pp.2345-2355.
[7] Aydin, O. and Yang, W.J., 2000. Natural convection in enclosures with localized heating from below and symmetrical cooling from sides. *International Journal of Numerical Methods for Heat & Fluid Flow*, *10*(5), pp.518-529.
[8] Nasr, K.B., Chouikh, R., Kerkeni, C. and Guizani, A., 2006. Numerical study of the natural convection in cavity heated from the lower corner and cooled from the ceiling. *Applied thermal engineering*, *26*(7), pp.772-775.
[9] Barletta, A., Nobile, E., Pinto, F., di Schio, E.R. and Zanchini, E., 2006. Natural convection in a 2D-cavity with vertical isothermal walls: Cross-validation of two numerical solutions. *International journal of thermal sciences*, *45*(9), pp.917-922.
[10] Dalal, A. and Das, M.K., 2005. Laminar natural convection in an inclined complicated cavity with spatially variable wall temperature. *International Journal of Heat and Mass Transfer*, *48*(18), pp.3833-3854.
[11] Snoussi, L.B., Chouikh, R. and Guizani, A., 2005. Numerical study of the natural convection flow resulting from the combined buoyancy effects of thermal and mass diffusion in a cavity with differentially heated side walls. *Desalination*, *182*(1-3), pp.143-150.
[12] Lewis, R.W., Nithiarasu, P. and Seetharamu, K.N., 2004. *Fundamentals of the finite element method for heat and fluid flow*. John Wiley & Sons.
[13] Reddy, J.N. and Gartling, D.K., 2010. *The finite element method in heat transfer and fluid dynamics*. CRC press.