NATURAL CONVECTION HEAT TRANSFER IN A SQUARE DUCT WITH V-CORRUGATED VERTICAL WALLS

By

MOHAMMAD ALI
B. Sc. Engg.(Mech.)

Thesis submitted to the Department of Mechanical Engineering in partial fulfillment of the requirement for the degree of

Master of Science

in

Mechanical Engineering.

April, 1990

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY, DHAKA,

BANGLADESH

RECOMMENDATION OF THE BOARD OF EXAMINERS

The Board of Examiners hereby recommends to the Department of Mechanical Engineering, Bangladesh University of Engineering and Technology, Dhaka, the acceptance of the thesis,"Natural Convection Heat Transfer In A Square Duct With V-Corrugated Vertical Walls", submitted by Mohammad Ali, in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering.

Chairman :

Dr. S. R. Husain
Assistant Professor
Deptt. of Mechanical Engineering
B.U.E.T., Dhaka.

Member :

Dr. S. M. Nazrul Islam
Professor and Head
Deptt. of Mechanical Engineering
B.U.E.T., Dhaka.

Member :

M. A. Taher Ali

Dr. M. A. Taher Ali
Professor
Deptt. of Mechanical Engineering
B.U.E.T., Dhaka.

Member :

Dr. Dipak Kanti Das
Professor
Deptt. of Mechanical Engineering
B.U.E.T., Dhaka.

Member (External) :

Khaliqur Rahman

Dr. Khaliqur Rahman
Professor
Deptt. of Chemical Engineering
B.U.E.T., Dhaka.

CERTIFICATE OF ORIGINALITY

This is to certify that the work presented in this thesis is an outcome of the investigation carried out by the author under the supervision of Dr. S. R. Husain, Assistant Professor, Department of Mechanical Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh.

Dr. S. R. Husain

Supervisor

Mohammad Ali

Author

ACKNOWLEDGEMENT

The author would like to express his sincere gratitude and respect to Dr. S. R. Husain, Assistant Professor, Department of Mechanical Engineering, Bangladesh University of Engineering and Technology (B.U.E.T.), Dhaka, for his continuous guidance, untiring supervision and constructive suggestions in performing this investigation that enabled the author to finish the study in its present form.

The author also expresses thankful gratitude to Dr. S. M. Nazrul Islam, Professor and Head, Department of Mechanical Engineering, B.U.E.T., Dhaka, for his valuable suggestions and kind permission for the use of microcomputer of this Department.

The author is highly grateful to Dr. Dipak Kanti Das, Professor, Department of Mechanical Engineering, B.U.E.T., Dhaka, for his suggestions and inspiration for completion of this thesis.

Sincere cooperation of Dr. M. A. Taher Ali, Professor, Department of Mechanical Engineering, B.U.E.T., Dhaka, for completion of this thesis is highly appreciable.

Special thanks are given to Dr. Khaliqur Rahman, Professor, Department of Chemical Engineering, B.U.E.T.,Dhaka,for his valuable suggestions.

The author is also indebted to Computer Center, B.U.E.T., Dhaka, for providing facilities without which this computational work would have not been possible.

Thanks are due to Mr. Abdus Salam for drawing the figures presented in this thesis and Mr. Fakrul Islam for his cooperation in typing this thesis.

ABSTRACT

A parametric study has been performed on natural convection heat transfer and flow characteristics in a square enclosure with V-corrugated vertical walls. The vorticity stream function formulation with the Control Volume based Finite Element Method (CVFEM) was used to analyze the effects of corrugation frequency and Grashof numbers on heat transfer and flow behaviour. The results show that the overall heat transfer through the enclosure decreases with increasing corrugation frequency for large Grashof numbers but the trend is reversed for low Grashof numbers. This behaviour can be explained as a manifestation of two competing effects : The increase of wall surface area versus the retardation of flow due to increase in corrugation frequency. Specifically the increase in wall surface area tends to enhance the overall heat transfer while the retardation of flow due to increased waviness tends to reduce the convective transport of energy.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF TABLES

NOMENCLATURE

Symbol	Meaning	Unit
u	Horizontal velocity	m/sec
v	Vertical velocity	m/sec
U	Dimensionless horizontal velocity	
V	Dimensionless vertical velocity	
x	Horizontal cartesian coordinate	m
y	Vertical cartesian coordinate	m
X	Dimensionless horizontal cartesian coordinate	
Y	Dimensionless vertical cartesian coordinate	
μ	Viscosity	kg/m-sec
ν	Kinematic viscosity	m^2/sec
ρ	Density	kg/m^3
ψ	Stream function	m^2/sec
Ψ	Dimensionless stream function	
ω	Vorticity	sec^{-1}
Ω	Dimensionless vorticity	
α	Thermal diffusivity, $k/\rho c_p$	m^2/sec
β	Coefficient of thermal expansion	$1/°k$
g	Acceleration due to gravity	m/sec^2
h	Convection heat transfer coefficient	$watt/m^2\ °K$
k	Thermal conductivity	watt/m °K

NOMENCLATURE (Continued)

Symbol	Meaning	Unit
Gr	Grashof number, $g\beta(T_h-T_c)L^3/\nu^2$	Dimensionless
Pr	Prandtl number, ν/α	Dimensionless
Ra_L	Rayleigh number, $g\beta(T_h-T_c)L^3/\nu\alpha$	Dimensionless

CHAPTER I

INTRODUCTION

A. Natural Convection Of Fluid.

The fluid flow in "free" or "natural" convection arises as a result of density variations caused by thermal expansion of the fluid in a non-uniform temperature distribution. Free convection currents transfer internal energy stored in fluid elements in the same manner as forced convection currents. However, the intensity of the mixing of the fluid is generally less in natural convection and consequently the heat transfer coefficients in natural convection are lower than those in forced convection. Since the temperature distribution is itself dependent on the movement of the fluid, the transport equations of motion and thermal energy are coupled. Further, the development of the flow is influenced by the shape of the heat transfer surfaces.

Both numerical and experimental methods have been used to obtain the solutions of heat transfer and fluid flow problems. Though experimental methods are more realistic, they are costly and time consuming due to the necessity of expensive prototypes and instrumentation. On the other hand, numerical methods can offer considerable savings in design time and costs. Of course, the validity of any numerical approach must be established by comparison with existing experimental data and other established numerical solutions before using it for simulating new problems. However, once tested and found to be reliable, it becomes a powerful tool for investigating a wide range of fluid flow and heat transfer problems. As a preliminary step, the work contained in this thesis consists of a numerical investigation of natural convection heat transfer and fluid flow within a square enclsure with V-corrugated vertical walls.

B. Background

This section is composed of two parts. In the first part, the studies that have been performed experimentally are discussed and the second contains a brief review of numerical investigations.

Experimental Investigations.

Several investigators have carried out their research on natural convection heat transfer and fluid flow with corrugated surfaces. First, experiments were performed for plane surfaces with different boundary conditions [1-8], and for corrugated surfaces [9-12]. Dropkin and Somercales [1] performed an experimental investigation on natural convection heat transfer in liquids confined by two parallel plates and inclined at various angles with respect to the horizontal. The range of Rayleigh numbers covered in these experiments was 5×10^4 to 7.17×10^8 and the Prandtl number was varied from 0.02 to 11560. Experiments were carried out in rectangular and circular containers having copper plates and insulating walls. The liquids used were water, silicon oil and mercury.

Imberger [2] made an experimental study on the natural convection heat transfer and fluid flow in closed cavities and obtained the solution of the Navier-Stokes equation with differentially heated end walls of the cavities of small aspect ratio and showed the strong agreement with the results of the same problem obtained numerically by Cormack et al. [4]. Ozoe et al. [8] performed a set of experiments for laminar flow in silicon oil and air along a rectangular channel. The channel was heated from below and cooled from above while the other two sides were insulated.

There have also been a number of investigations carried out on heat transfer problems with V-corrugated surfaces. Chinnappa [9] carried out an experimental investigation on natural convection heat transfer from a horizontal lower hot V-corrugated plate to an upper cold flat plate. He took data for a range of Grashof numbers from 10^4 to 10^6. The author noticed a change in the flow pattern at Gr = 8 X 10^4, which he concluded as a transition point from laminar to turbulent flow. In his work, Chinnappa found that for horizontal air layers the enclosure ends had no effect on heat transfer within the range of experimental variables.

Elsherbiny et al. [10] investigated free convection heat transfer for air layers bounded by a lower hot V-corrugated plate and an upper cold flat plate. A single correlation equation in terms of Nusselt number, Rayleigh number, tilt angle, aspect ratio was developed for aspect ratio ranging from 1 to 4, angle of inclination ranging from 0 to 60 degrees and Rayleigh number ranging from 10 to 4 X 10^6. They claimed that the convective heat transfer across air layers bounded by V-corrugated and flat plates was greater than those for two parallel flat plates by a maximum of 40%.

Randall et al. [11] studied local and average heat transfer coefficients for natural convection between a V-corrugated plate (60° V-angle) and a parallel flat plate using interferometric techniques to find the temperature distribution in the enclosed air space. From this temperature distribution they used the wall temperature gradient to estimate the local heat transfer coefficient. Local values of heat transfer coefficient were investigated over the entire V-corrugated surface area. The author recommended a correlation in which the heat flux of 10% is higher than that for parallel flat plates.

An experimental investigation of heat transfer by natural convection from an inclined hot sinusoidal corrugated plate at the bottom to an inclined cold flat plate at the top in a bounded rectangular region was carried out by Kabir [12]. The vertical side walls of the enclosure were plane and adiabatic. By comparing with other related works it was concluded that for the same plate spacing the heat transfer rates across air layers bounded by the corrugated and flat plate were greater than those for two parallel flat plates by a maximum of 40%.

Numerical Investigations

Natural convection heat transfer from a plane surface with different boundary conditions has been studied numerically by several researchers. Zhong et al. [13] carried out a finite-difference study to determine the effects of variable properties on the temperature and velocity fields and the heat transfer rate in a differentially heated, two dimensional square enclosure.

Nayak et al. [14] considered the problem of free and forced convection in a fully developed laminar steady flow through vertical ducts under the conditions of constant heat flux and uniform peripheral wall temperature. Chenoweth et al. [15] obtained steady-state, two dimensional results from the transient Navier-Stokes equations given for laminar convective motion of a gas in an enclosed vertical slot with large horizontal temperature differences. Sofir Uddin et al.[16] investigated the natural convection heat transfer and fluid flow behaviour for vertical sinusoidal walls. The results showed that for corrugation frequency=3 with different Grashof numbers the total heat flux becomes lower than that for straight wall with corresponding Grashof numbers and for corrugation frequency=1 the total heat flux becomes higher than that for straight one.

C. Motivation of the Present Investigation.

The study of Natural Convection effects is important in numerous engineering applications. In designing nuclear reactors, solar collectors, electrical and microelectronic equipment containers and in many other designing problems, natural convection heat transfer is prominent. Thus, for different boundary conditions and shapes the analysis of the effects of natural convection is necessary to ensure efficient performance of the various heat transfer equipment. Several investigators [8-10,12] performed their studies on convection heat transfer with corrugated walls experimentally, but they considered a horizontal lower hot corrugated to an upper cold flat plate only. None of them performed an experiment on convection heat transfer with vertical hot and cold corrugated plates. However, there is no knowledge of numerical simulation of natural convection heat transfer and fluid flow with V-corrugated vertical walls, which forms the basis for the motivation behind the present study.

D. Objectives of the Study.

The main objective of this thesis is to numerically simulate heat transfer and fluid flow behaviour inside a square enclosure with V-corrugated vertical walls and insulated horizontal walls. Specifically, the coupled momentum and energy transport equations will be solved with the Grashof number and corrugation frequency as parameters. The effects of corrugation frequency and Grashof number on local and overall heat transfer rates, velocity and temperature distribution will be examined both qualitatively and quantitatively. The effect of increasing the corrugation frequency will lead to a greater heat transfer surface but whether the overall heat transfer rate will increase or decrease is an important question which will be addressed in the analyses of the results.

CHAPTER II

NATURAL CONVECTION IN A SQUARE DUCT WITH VERTICAL V-CORRUGATED WALLS.

A. Problem Statement

The Problems of Natural Convection Heat Transfer and Fluid Flow in a square enclosure are considered in this analysis. The fluid flow is caused by the buoyant force which is the consequence of temperature gradients inside the enclosure. The temperature field itself is described by the transport equation for energy. The problem is therefore described by a coupled set of momentum and energy equations.

The problem schemetic is shown in Fig.1. The top and bottom walls of the enclosure are insulated and the left and right vertical walls are V-corrugated. The left and right walls are kept at constant temperature. The temperature of the left wall is T_h and that of the right wall is T_c, where $T_h > T_c$. The characteristic length of the square enclosure is L. The origin of the X-Y coordinate system is located at the left-bottom corner of the cavity.

B. Governing Equations and Boundary Conditions.

Governing Equations

The Navier-Stokes equations for two-dimensional, incompressible flow with constant properties in cartesian coordinates can be written as follows:

Continuity equation,

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

x-momentum equation,

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + S^u \tag{2}$$

y-momentum equation,

$$u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) + S^v \tag{3}$$

In the above equations, u and v represent the velocity components in the x and y directions respectively and p is the pressure. The source terms s^u and s^v consider the other body and surface forces in the x and y directions respectively and ν is the kinematic viscosity.

By differentiating equations (2) and (3) with respect to y and x respectively and then subtracting the results of the former from the latter, a single vorticity transport equation can be obtained:

$$u\frac{\partial \omega}{\partial x} + v\frac{\partial \omega}{\partial y} = \nu\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right) + \left(\frac{\partial S^v}{\partial x} - \frac{\partial S^u}{\partial y}\right) \tag{4}$$

where ω is the vorticity defined as

$$\omega \equiv \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \tag{5}$$

Upon defining the streamfunction, ψ as

$$\frac{\partial \psi}{\partial y} \equiv u \tag{6}$$

$$-\frac{\partial \psi}{\partial x} \equiv v \tag{7}$$

the Poisson equation relating ω to ψ may be obtained by substituting (6) and (7) into (5):

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} + \omega = 0 \tag{8}$$

The equations (4) and (8) are equivalent to equations (1), (2) and (3) where instead of two momentum equations, a single transport equation for vorticity is revealed and the pressure gradient terms are absent.

Assuming the properties to be constant other than the density variation in the buoyant forces, the Boussinesq approximation [18] may be used on equation (4) which results in

$$u\frac{\partial \omega}{\partial x} + v\frac{\partial \omega}{\partial y} = \nu\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right) + g\beta\frac{\partial T}{\partial x} \tag{9}$$

The energy transport equation for two dimensional incompressible flow with constant properties can be written as

$$u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} = \alpha \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) \tag{10}$$

where α is the thermal diffusivity of the fluid.

Equations (4) to (10) can be normalised by introducing the following non dimensional quantities:

$$X \equiv \frac{x}{L} \tag{11}$$

$$Y \equiv \frac{y}{L} \tag{12}$$

$$U \equiv \frac{uL}{\nu} \tag{13}$$

$$V \equiv \frac{vL}{\nu} \tag{14}$$

$$\Omega \equiv \frac{\omega L^2}{\nu} \tag{15}$$

$$\Psi \equiv \frac{\psi}{\nu} \tag{16}$$

$$\theta \equiv \frac{T - T_c}{T_h - T_c} \tag{17}$$

to yield

$$U \frac{\partial \Omega}{\partial X} + V \frac{\partial \Omega}{\partial Y} = \frac{\partial^2 \Omega}{\partial X^2} + \frac{\partial^2 \Omega}{\partial Y^2} + Gr \frac{\partial \theta}{\partial X} \tag{18}$$

$$U \frac{\partial \theta}{\partial X} + V \frac{\partial \theta}{\partial Y} = \frac{1}{Pr} \left(\frac{\partial^2 \theta}{\partial X^2} + \frac{\partial^2 \theta}{\partial Y^2} \right) \tag{19}$$

where Gr and Pr are the Grashof and Prandtl numbers, respectively, and defined as :

$$Gr = g\beta(T_h - T_c)L^3/\nu^2 \tag{20}$$

$$Pr = \nu/\alpha \tag{21}$$

The dimensionless auxiliary equations are :

$$\Omega = \frac{\partial V}{\partial X} - \frac{\partial U}{\partial Y} \tag{22}$$

$$U = \frac{\partial \Psi}{\partial Y} \tag{23}$$

$$V = -\frac{\partial \Psi}{\partial X} \tag{24}$$

Here, the parameters g, ß and α represent the acceleration due to gravity, the coefficient of thermal expansion, and the thermal diffusivity of the fluid respectively.

Boundary Conditions

The boundary conditions of the problem are as follows :

(i) $U = V = 0$ at all walls

(ii) $\Psi = 0$ at all walls

(iii) $\theta = 1$ at left wall

$\theta = 0$ at right wall

(iv) $\left(\frac{\partial \theta}{\partial Y}\right)_{Y=0} = \left(\frac{\partial \theta}{\partial Y}\right)_{Y=1} = 0$

C. **Method of Solution.**

The calculation domain is first divided into quadrilaterals by a number of vertical and horizontal grid lines. These in turn are sub-divided into triangular elements (an example of discretization of the domain is shown in Fig.2). On each element, links are constructed by joining the mid-points of the sides to the centroid. These links come into contact with other adjacent elements to form closed regions called control volumes (shown in Fig.3).

Following the domain discretization, the integral formulation of the relevant transport equation is imposed on each control volume of the overall region. This is done by prescribing a suitable shape function within each element which is used to express the combined convective-diffusive flux variation along the links of same. These fluxes are integrated and the contributions of the links to the control volume portions are assembled in a systematic manner. This procedure is repeated for all the elements in the calculation domain. The net outcome is a set of nodal equations for the transported variable ϕ, which may be written as:

$$a_p^{\phi}\ \phi_p + \sum_{nb} a_{nb}^{\phi}\ \phi_{nb} = b_p^{\phi} \tag{25}$$

where nb denotes the neighbour nodes of the node p, a are the coefficients of the Nodal Equation Matrix (NEM) for node p, and b is the Global Load Vector (GLV) component corresponding to node p. The details involved in obtaining equation (25) are available in [17].

The solution of the system of equations (25) is obtained iteratively in as much that the coefficients (a) themselves depend on the values of ϕ, where ϕ can represent either the vorticity, streamfunction, or temperature. The procedure adopted in this investigation is due to [19] and is summarized as follows :

1. Compile the coefficients for the poisson equation (8) for the stream function. These coefficients do not change from one iteration to the next.
2. Guess the distribution of streamfunction Ψ and compile the coefficients of the vorticity transport equation.
3. Solve for the vorticity Ω and update the values as proposed in [17] and [19].
4. Using the values of Ω obtained in step(3), compile the GLV for the Poisson equation for Ψ.
5. Solve for streamfunction using the coefficients from (1)
6. Check the convergence as per suggestion in [19].
7. If not converged, go to step (2).
8. If converged, perform post processing tasks such as heat transfer calculation.

Once convergence of the governing equation has been achieved, the following quantities are calculated :

(i) The local Nusselt number along the hot wall, Nu_Y,

$$Nu_Y \equiv \frac{\dot{q}''L}{k(T_h-T_c)} = -\frac{\partial\theta}{\partial N} \tag{26}$$

(ii) The dimensionless total heat flux at the hot wall,

$$Q \equiv -\int_{Y=0}^{Y=1} \frac{\partial\theta}{\partial N}\, dS(Y) \tag{27}$$

where S is the dimensionless distance measured along the corrugation of the wall and N is the dimensionless distance measured normal to same.

In equations (26) and (27), $\dot{q}''$ is the heat flux rate per unit length at the hot wall and k is the thermal conductivity.

CHAPTER III

RESULTS AND DISCUSSION

A parametric study was conducted to analyze the effects of corrugation frequency and Grashof number on natural convection heat transfer and fluid flow inside a square enclosure with V-corrugated vertical walls. The discussion of the results that follows are those obtained by using a 31 X 31 mesh. A grid refinement study was made using a 49 X 49 mesh for the highest Grashof number and corrugation frequency ($Gr = 10^5$, C.F.= 3) and it was found that the results from the 31 X 31 grid runs was accurate to within about 3 percent. The comparison of the flow fields can be made by referring to Fig. (4-8) and the overall heat transfer to Table-3 where it is seen that the 31 X 31 mesh results agree quite well with those of the 49 X 49 mesh.

In this investigation the total heat transfer through the enclosure, vertical velocity and temperature distributions at the horizontal mid-plane and local Nusselt number along the hot wall were examined with respect to Grashof numbers 10^3, 10^4, 10^5 and corrugation frequencies 1,2,3. The corrugation amplitude was fixed at 5 percent of the enclosure height for all runs, where the amplitude " A " is defined as half the horizontal distance, measured from the left extremity of the left wall to its right extremity (see Fig. 1). Henceforth, the left and right extremities of the hot wall will be referred to as the "trough" and "peak", respectively. The local and total heat flux with respect to corrugated and straight walls and flow characteristics with respect to grid refinement were also compared. The summary of computational runs has been shown in Table-1. All results are represented in dimensionless form.

TABLE - 1.

Summary of Computational Runs

Corrugation Frequency (C.F.)	Grashof numbers	Grid Size
	10^3	31 X 31
0 (Straight Wall)	10^4	Do
	10^5	Do
	10^3	31 X 31
1	10^4	Do
	10^5	Do
	10^3	31 X 31
2	10^4	Do
	10^5	Do
	10^3	31 X 31
3	10^4	Do
	10^5	Do
	10^3	49 X 49
3	10^4	Do.
	10^5	Do

A. Effect Of Corrugation Frequency.

Table-2 shows the effects of Corrugation Frequency (C.F.) on total heat flux with different Grashof numbers. In Table-2, the increase of C.F. from 0 to 1 leads to higher value of Q for all Grashof numbers which may be attributed to the enhancement of surface area. However, for Grashof number (Gr) 10^4 and 10^5, Q decreases with increase in C.F. from 1 to 3, but increases continuously for Gr = 10^3. This behaviour may be explained by asserting that at high Grashof numbers the fluid velocity increases near the peaks but drops near the troughs as the boundary layer tends to separate. Thus the fluid fails to maintain close contact near the troughs of the corrugation, resulting in decreased convection heat transfer, whereas for Gr = 10^3, the low vertical velocities thus generated enable the fluid to maintain better contact with the corrugated wall. Thus with increasing C.F. the corresponding enhancement of heat transfer surface area leads to increased total heat flux at low Gr, but for the case of high Gr, the lower velocities and consequent decrease in convective heat transfer at the troughs more than offsets the increased surface area. This decrease in convection heat transfer is evident upon referring to Fig. 9 and Fig. 10, where it may be observed that the local Nusselt number attain minimum values at the troughs of the corrugation.

Fig. 9 Fig. 10 and Fig. 11 indicate the effects of C.F. on local Nusselt number along the hot wall with the Grashof number as parameter. The local Nusselt number is identical to the dimensionless local heat flux. It can be noted from these figures that there is a significant increase in local Nusselt number at the peaks of the corrugation and decrease of the same at the troughs. The reason for this is that the peaks cause the fluid to come in contact more intimately with the surface resulting in large convection heat transfer and consequently the local Nusselt number increases. Another observation that may be made in table-2 is that at C.F.=3, Q is less than that for C.F.=0 with $Gr = 10^4$ and 10^5. This indicates that vertical V-corrugation can be used to reduce the heat transfer through the enclosure, provided that sdthe corrugation frequency is sufficiently large.

Referring to Figures 9,10,11 again, it is seen that the peak values of Nu_Y decreases with increasing vertical distance along the corrugated wall. This may be explained by the fact that the colder fluid collects at the bottom-left corner of the enclosure creating a large temperature gradient with the hot wall, which is the main driving force for heat transfer at the wall and as it moves up and receives heat, the temperature gradient decreases, causing the decrease in local Nusselt number.

Fig. 12 and Fig. 13 reveal the effect of C.F. on vertical velocity distribution at the horizontal mid-plane for Grashof number 10^5 and 10^3 respectively. Fig. 12 indicates that the peak value of the vertical velocity decreases with increase in C.F. This trend can be explained by examining Fig. 14, which indicates that the temperature gradient is lower for higher C.F., causing a lower buoyant force and hence a lower vertical velocity. Because of this lower velocity, the strength of convection heat transfer decreases with increasing C.F. which is shown in Table-2. But in Fig. 13 the vertical velocity increases with C.F., which leads to an increase in overall heat transfer.

TABLE - 2.

The Variation of the Total Heat Flux for Corrugated and Straight Walls with Different Grashof Number.

Corrugation Frequency (C.F.) / Grashof number(Gr)	C.F=1.	C.F=2.	C.F=3.	Straight wall
Gr = 10^3	1.126	1.132	1.135	1.121
Gr = 10^4	2.295	2.271	2.238	2.270
Gr = 10^5	4.837	4.753	4.573	4.724

B. Effect of Grashof Number

Fig. 15 shows the variation of Q as a function of Gr with C.F. as the parameter. It may be seen here that the variation of Q with C.F. is greater for higher Gr. Further, the different curves for Q vs. Gr "cross" at around $Gr = 10^3$ indicating a trend reversal which was discussed earlier. In this connection, attention is drawn to Fig.16,17 and 18 which show the variation of temperature along the horizontal mid-plane for different Grashof numbers. It is clearly evident from these figures that for $Gr=10^3$, the temperature decreases linearly as one proceeds from the left to the right wall of the enclosure, which indicates that the heat transfer is primarily dominated by conduction. This further substantiates the trend of increasing Q with C.F. for low Grashof numbers by conduction from the increased surface area, where fluid flow retardation by increasing corrugation is not significant.

TABLE-3.

Total Heat Flux for Grids 31 X 31 and 49 X 49
Corrugation Frequency = 3.

Grashof number (Gr) / Grid Size	$Gr=10^3$	$Gr=10^4$	$Gr=10^5$
31 X 31	1.1347	2.2376	4.5724
49 X 49	1.1345	2.2222	4.4765

CHAPTER IV

CONCLUSIONS

A. Summary of Results

In this investigation the effects of corrugation frequency and Grashof number on the local and total heat flux and flow characteristics have been observed and discussed. The total heat flux for different corrugation frequency with Grashof number has been compared with the straight wall. The overall heat transfer rate through the enclosure was found to vary little with change in corrugation frequency, but the local heat flux rate displayed large changes along the corrugated walls. In particular, for low Grashof number, the overall heat flux rate increased continuously with corrugation frequency whereas the trend was reversed for higher Grashof numbers. It can therefore be concluded that there are two competing phenomena that give rise to the variation in total heat flux: (1) The enhancement of Q due to increasing surface area and (2) The decrease of Q due to flow retardation by increasing corrugation. Specifically, at low Gr, the conduction mode prevails and as C.F. increases, Q is enhanced, whereas for high Gr, the fluid fails to collect heat by transport due to increasing corrugation. This trend may find application in practical situations where heat transfer reduction is desired across large temperature differences by increasing the corrugation of the vertical wall to the point where Q is less than that for the straight wall case.

B. Proposal of Further Work.

This has been a preliminary study and experimental results are necessary to corroborate the numerical results presented herein. Furthermore, due to computational limitations the effects of higher C.F, corrugation amplitude and variation of enclosure aspect ratio were not looked into. Also, the transient nature of the flow can be investigated. It is possible that at very high Grashof number, the system may become temporally oscillatory,and a transient solution can help predict this behaviour. Another extension that may be made is to calculate the flow field for very large Grashof numbers and turbulent flow using a two equation turbulence transport model, such as the $K-\epsilon$ model.

Further, only two dimensional heat transfer and fluid flow problem has been analyzed in this thesis. So this deliberation may be extended to three dimensional analysis to investigate the effects of the end surfaces on heat transfer and flow field. In addition, the problem of heat transfer and fluid flow along a corrugated surface in an infinite fluid may be studied to examine the boundary layer behaviour.

REFERENCES

1. Dropkin, D., and Somercales, E., " Heat Transfer by Natural Convection in Liquids Confined by Two Parallel Plates Which are Inclined at Various Angles with Respect to the Horizontal ", J. Heat Transfer. Series c, Vol. 87, 1965, pp. 77 - 84.

2. Imberger, J., " Natural Convection in a Shallow Cavity with Differentially Heated End Walls: Experimental Solutions ", J. Fluid Mechanics, Vol. 65, 1974, Part 2, pp. 247-260.

3. Hart, J., " Stability of the Flow in a Differentially Heated Inclined Box", J. Fluid Mechanics, Vol. 47, 1971, pp. 547-576.

4. Cormack, D.E., Leal, L.G., and Seinfeld, J.H., " Natural Convection in a Shallow Cavity with Differentially Heated End Walls: Numerical Solutions ", J. Fluid Mechanics, Vol.65, 1974, Part 2, pp. 231 - 246.

5. Inaba, H., and Fukuda, T., " Natural Convection in an Inclined Square Cavity in Regions of Density Inversion of Water", J., Fluid Mechanics, Vol. 142, 1984, pp. 363 - 381.

6. De Graaf, J. G. A., and Vander Held, F. E. M., "The Relation between the Heat Transfer and Convection Phenomena in Enclosed Plain Air Layers", Appl. Sci. Res. A. Vol. 3, 1953, pp. 393 - 409.

7. Globe, S., and Dropkin, D., "Natural Convection Heat Transfer in Liquids Confined by Two Horizontal Plates and Heated from Below", J. Heat Transfer Vol. 81, Series.c, 1959, pp. 24 -28.

8. Ozoe, H., Sayma, H., and Churchill, S . W., " Natural Convection in an Inclined Rectangular Channel at Various Aspect Ratios and Angles : Experimental Measurements ", Int. J. Heat & Mass Transfer, Vol. 18, 1975, pp. 1425-1431.

9. Chinnappa, J.C.V., " Free Convection in Air between a 60° Vee-Corrugated Plate and Flat Plate ", Int. J. Heat & Mass Transfer, Vol. 13, 1970, pp. 117 - 123.

10. Elsherbiny, S.M., Hollands, K.G.T., and Raithby, G.D., " Free Convection Across Inclined Air Layers with One Surface V -Corrugated", in Heat Transfer Solar Energy Systems, Howell, J.R. and Min, T., (eds.) ASME, New York, 1977.

11. Randall, R.K. et al. "Interferometric Investigation of Convection in Salt-Flat Plate and Vee-Corrugated Solar Collectros", Solar Energy International Progress,pp. 447-460.

12. Kabir, H., "An Experimental Investigation on Natural Convection Heat Transfer from a Hot Corrugated Plate to a Flat Plate", M.Sc. Engg. Thesis, 1988, BUET.

13. Zhong, Z. Y., Yang, K. T., and Lloyd, J.R., "Variable Property Effects in Laminar Natural Convection in a Square Enclosure", J. Heat Transfer, Vol. 107, 1985, pp.133-146.

14. Nayak, A. L., and Cheny, P.,"Finite Element Analysis of Laminar Convection Heat Transfer in Vertical Ducts with Arbitrary Cross Section", Int. J. Heat & Mass Transfer Vol. 18, 1975, pp. 227 - 236.

15. Chenoweth, D.R., and Paolucci, S. " Natural Convection in an Enclosed Vertical Air Layer with Large Horizontal Temperature Differences", J. Fluid Mechanics, Vol.169, 1986, pp. 173 - 210.

16. Sofir Uddin, M., Al-Azad, S., and Roy, B.D., "Natural Convection Heat Transfer and Fluid Flow in a Square Enclosure with Corrugated Vertical Walls", B.Sc. Engg. Thesis, 1988, BUET.

17. Husain, S. R., " Extensions of the Control Volume based Finite Element Method for Fluid Flow Problems", Ph.D. Thesis, Texas A & M University, 1987.

18. Turner, J. S., "Buoyancy Effects in Fluids".

19. Patankar, S. V., " Numerical Heat Transfer and Fluid Flow ", Washington, D.C. Homisphere, 1980.

20. Simpkins, P.G., and Chen, K. S., " Convection in Horizontal Cavities", J. Fluid Mechanics, Vol. 166, 1986, pp. 21 - 39.

21. Kreith, F., "Principles of Heat Transfer".

22. Yuan, S. W., "Foundation of Fluid Mechanics".

23. Baliga, B. R., and Patankar, S. v., "A New Finite Element Formulation for Convection-Diffusion Problems", Numer, Heat Transfer, Vol.3, pp. 393-409, 1980.

APPENDIX - A

FIGURES

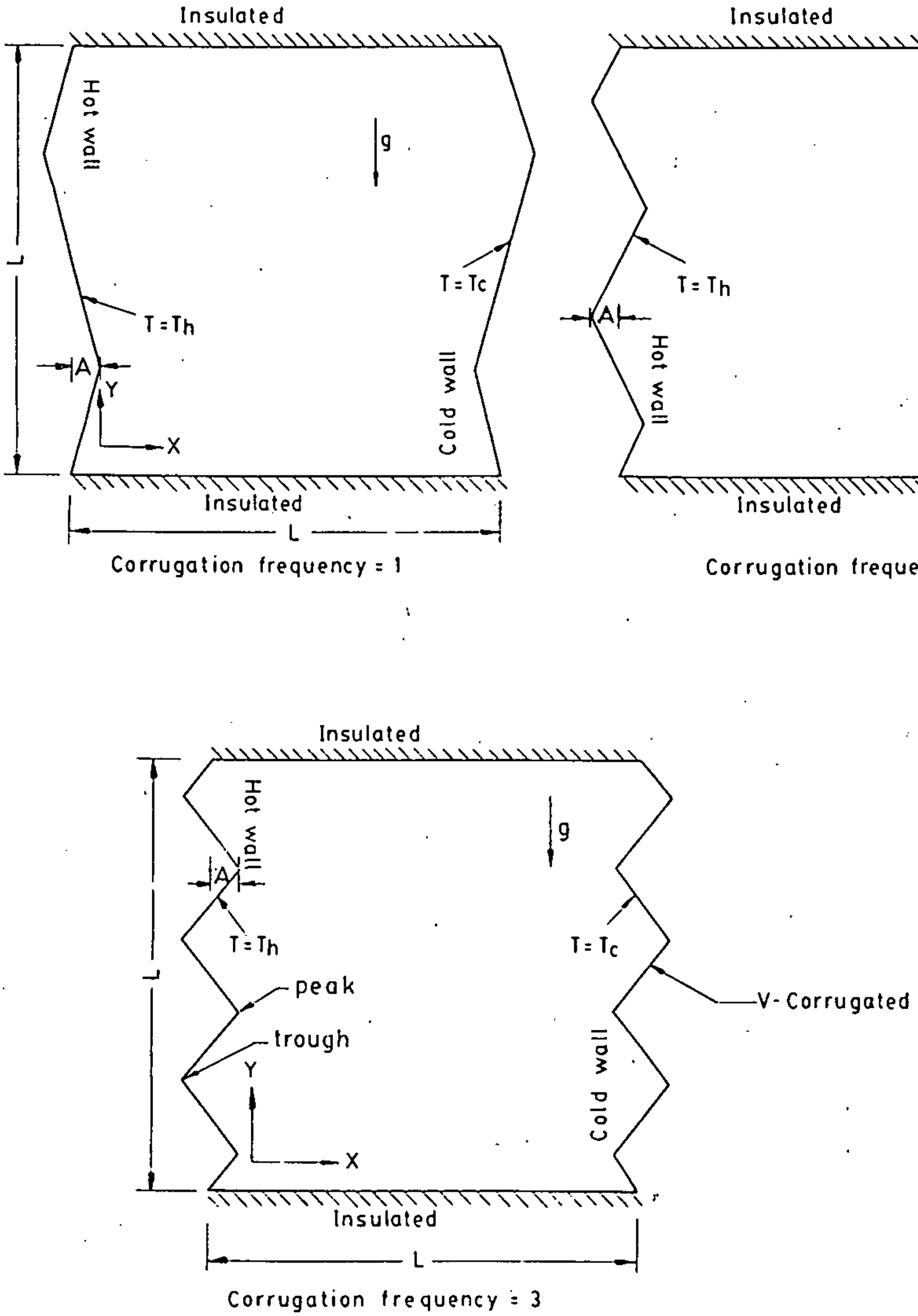


Fig. 1. Schematic of the Calculation Domain.

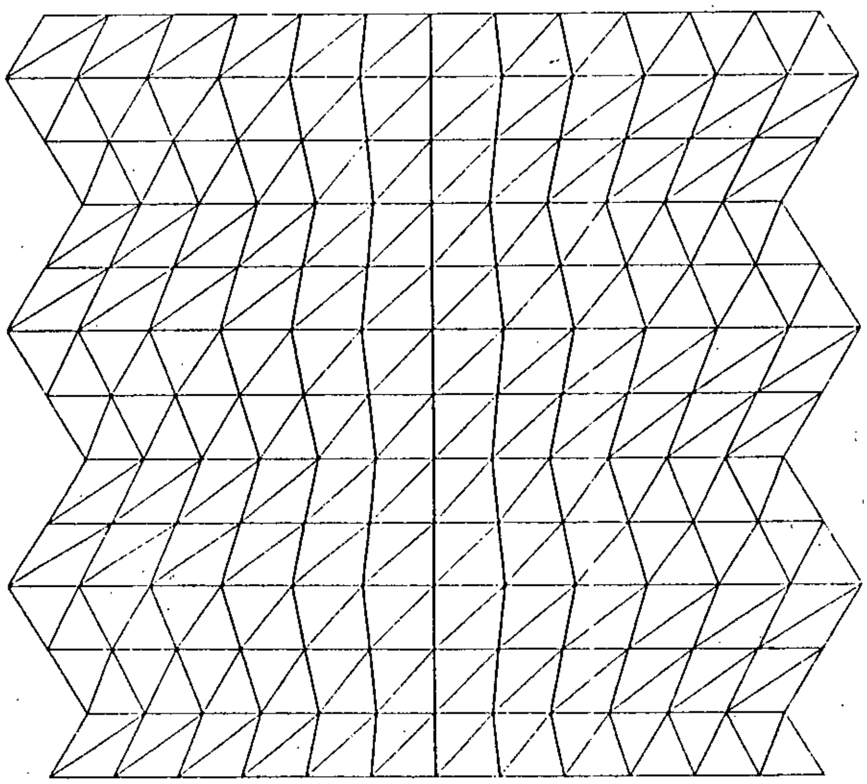


Fig. 2. Domain Discretization.

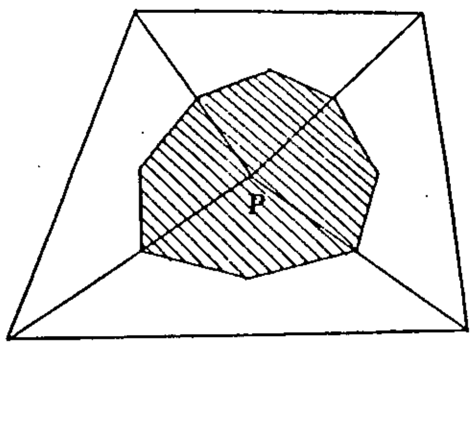


Fig. 3. A Typical Control Volume Shown Shaded.

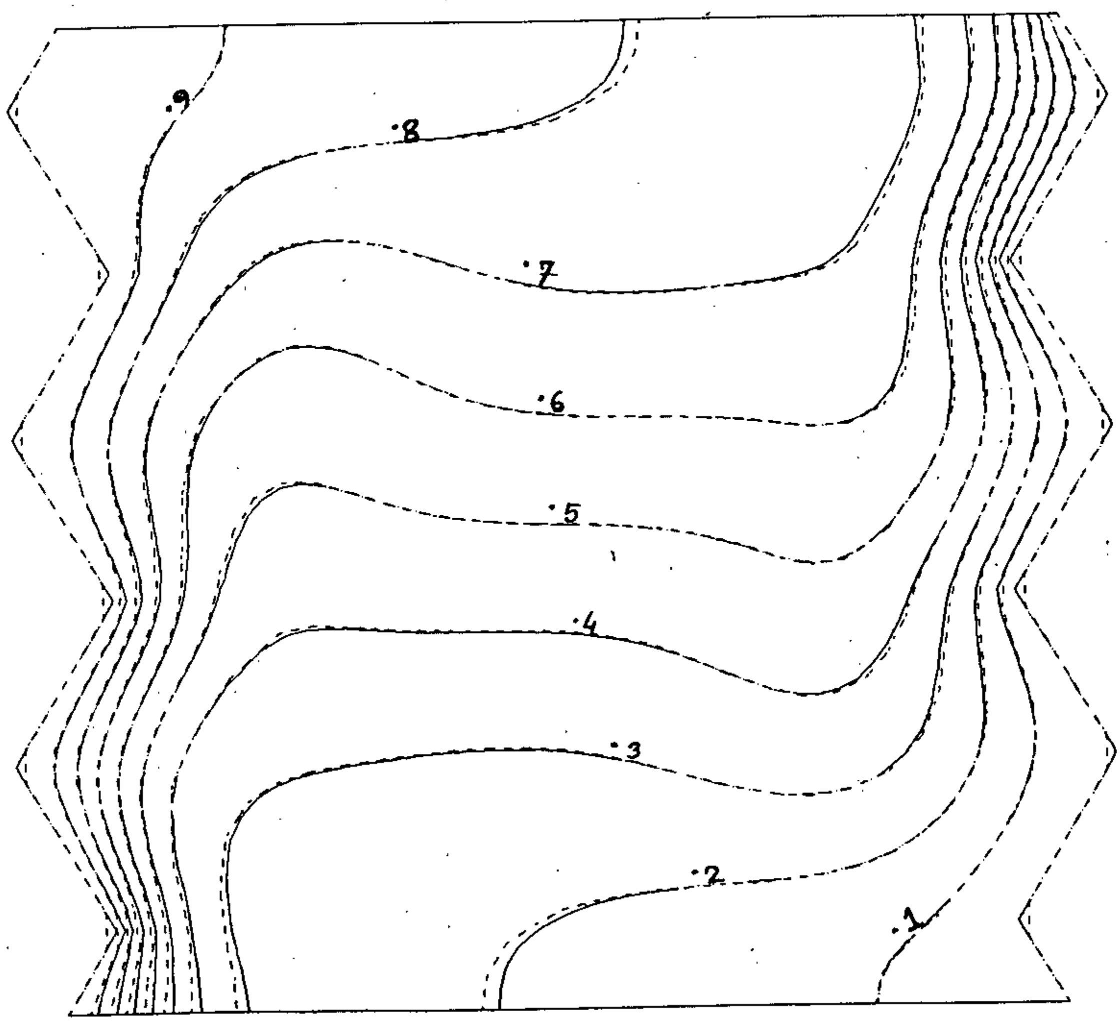


Fig. 4 Isotherm Plot for Different Grid Size.
$Gr = 10^5$ C.F.= 3
———————— (49 X 49)
- - - - - - - - - - - (31 X 31)

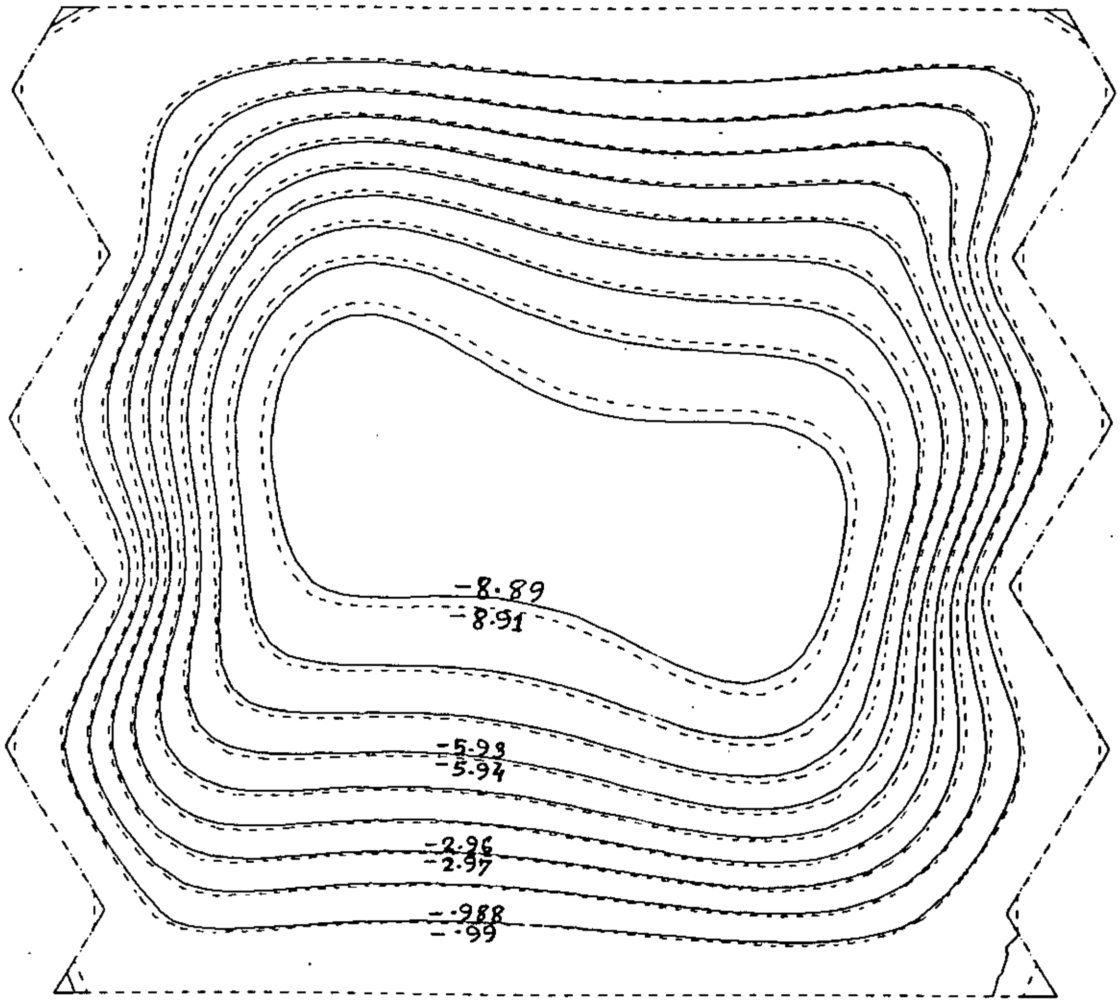


Fig. 5 Stream Line Plot with Grid Refinement

$Gr = 10^5$ & C.F.= 3

———————— (49 X 49)

---------- (31 X 31).

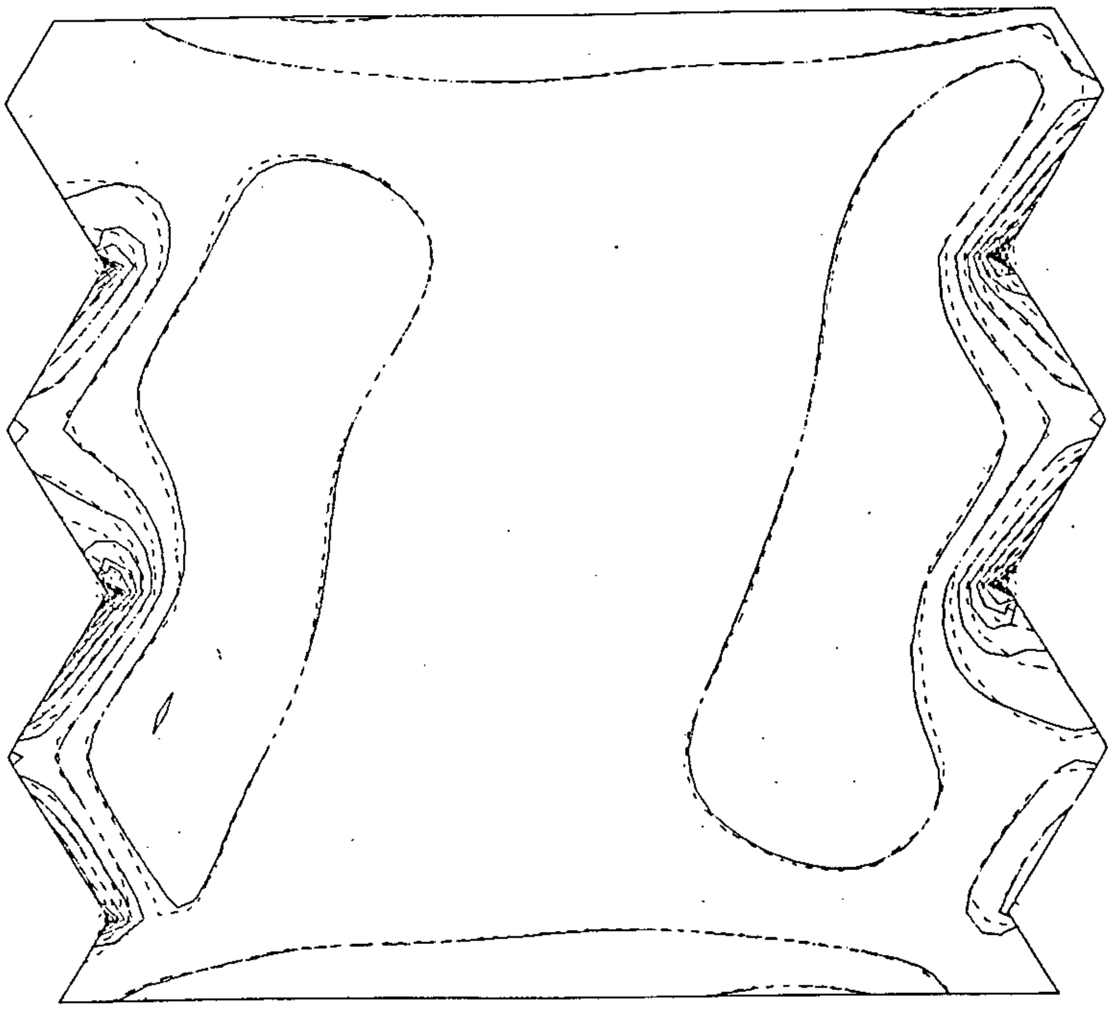


Fig. 6 Vorticity Contour Plot with Grid Refinement.

Gr = 10^5 C.F. = 3

———————— (49 X 49)

------------ (31 X 31)

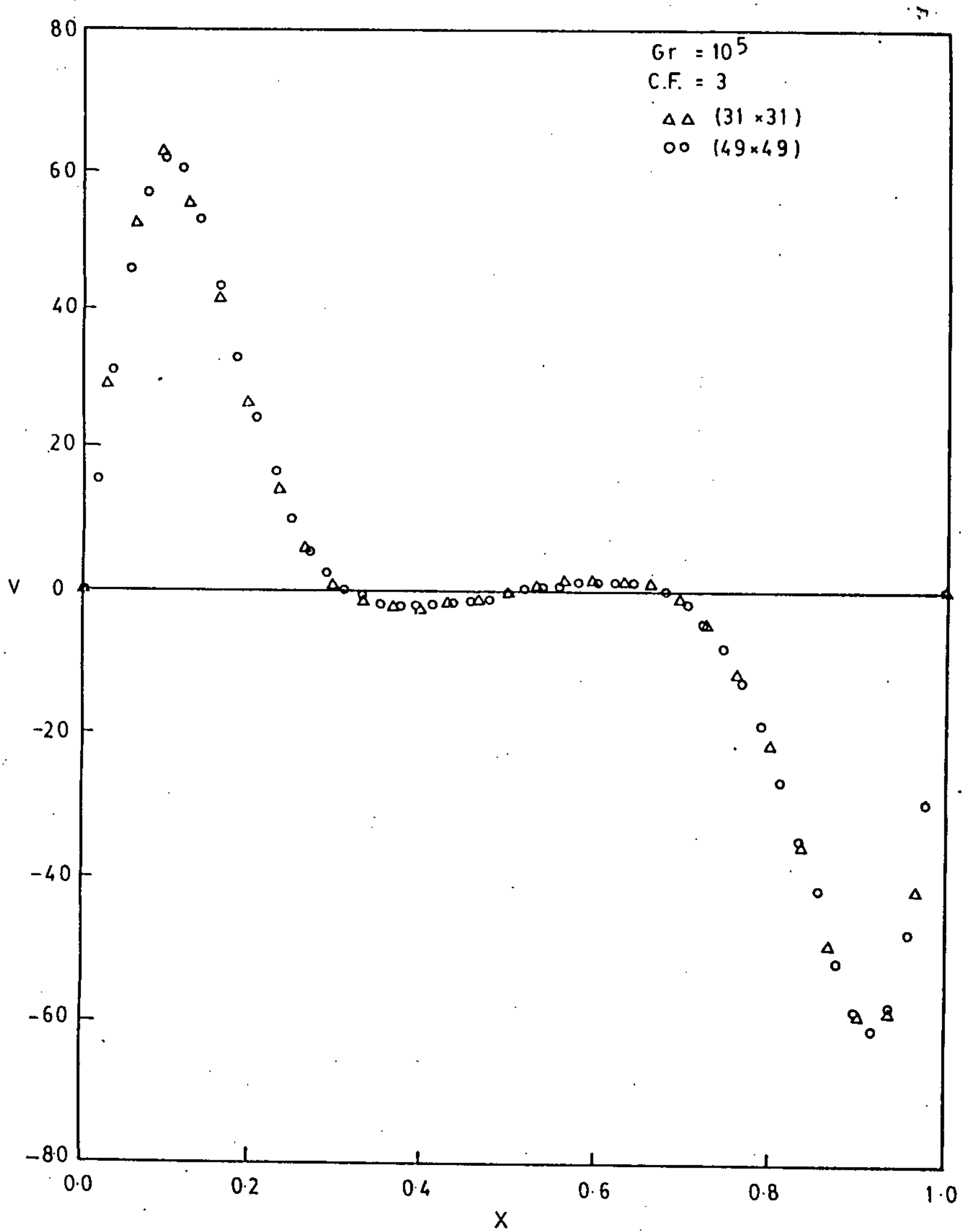


Fig. 7 Vertical Velocity Distribution at the Horizontal Mid Plane for Different Grid Size.

Gr = 10^5 & C.F. = 3

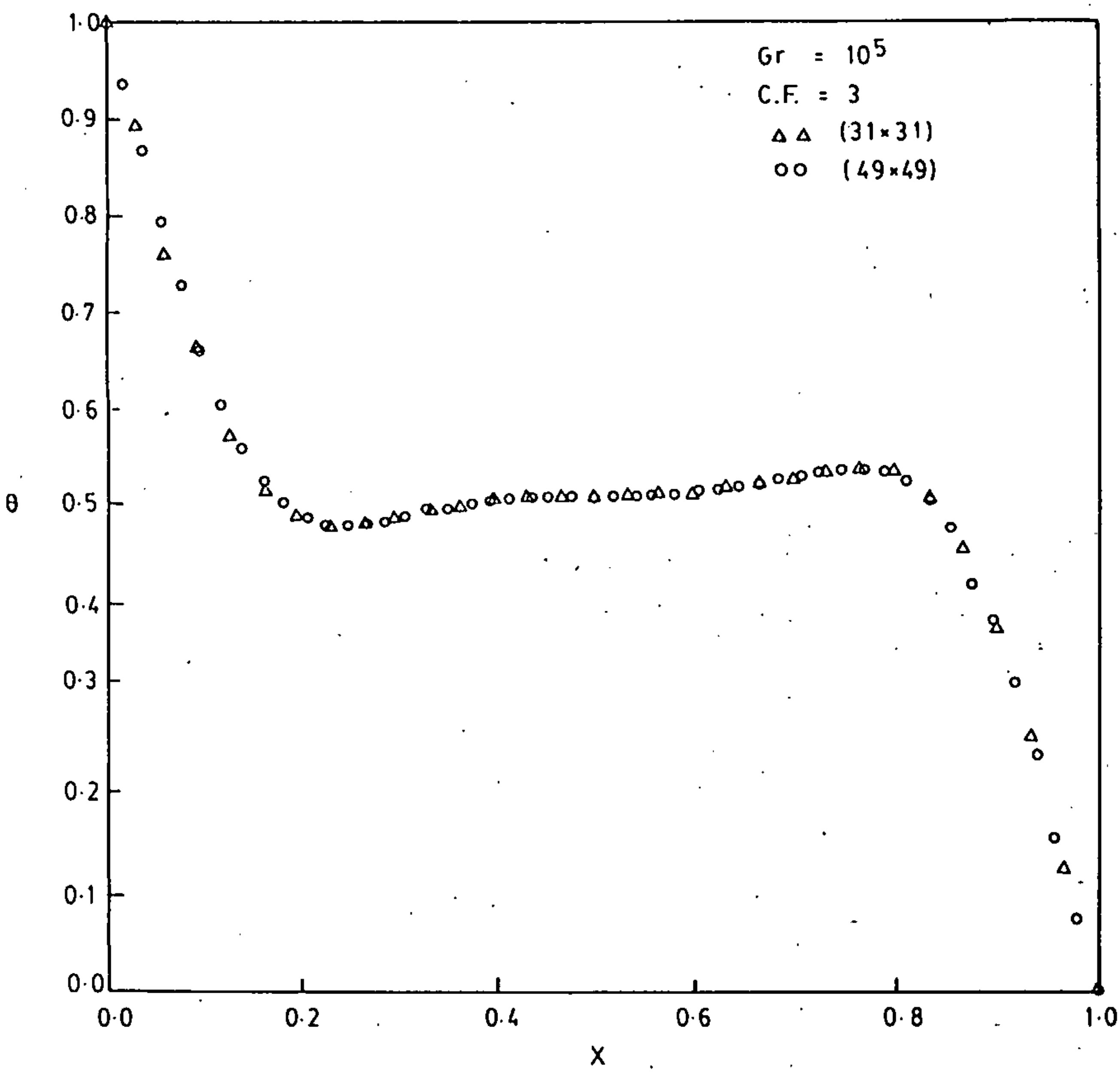


Fig. 8 Temperature Distribution at the Horizontal Mid Plane for Different Grid Size.

$Gr = 10^5$ & C.F. = 3

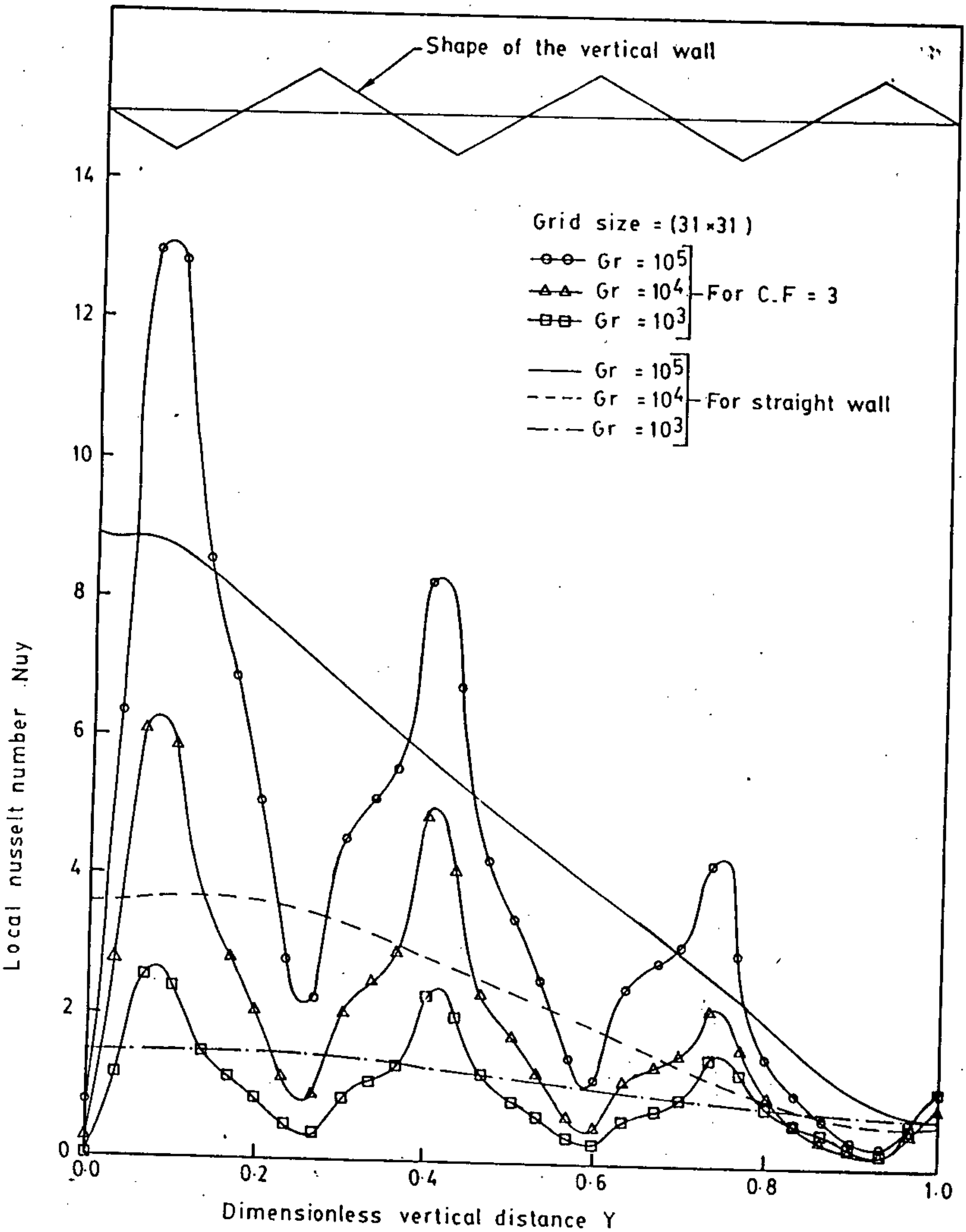


Fig. 9 Local Nusselt Number Distribution on the Hot Wall.

C.F. = 0 & 3.

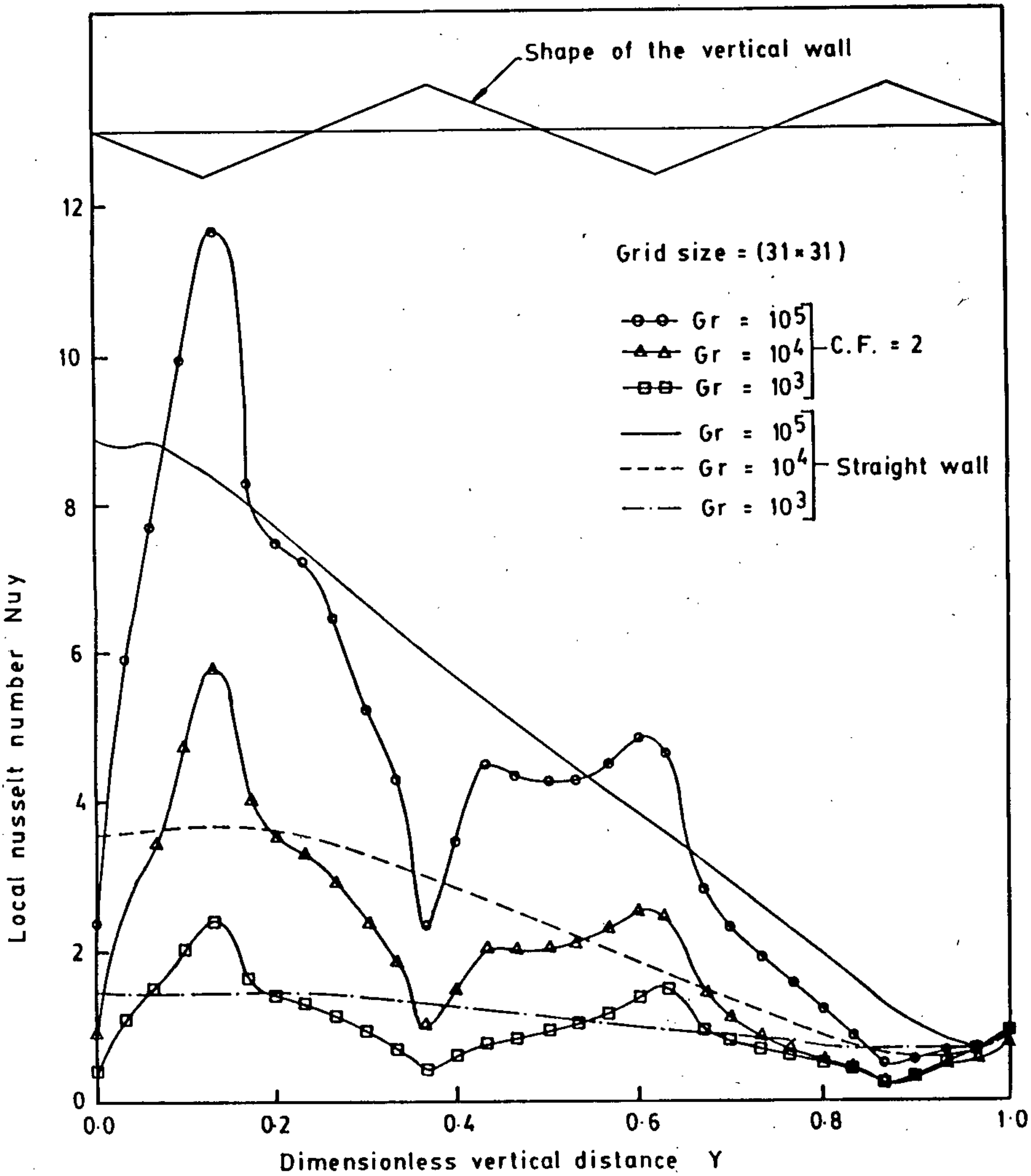


Fig. 10 Local Nusselt Number Distribution on the Hot Wall.

C. F. = 2 & Straight Wall.

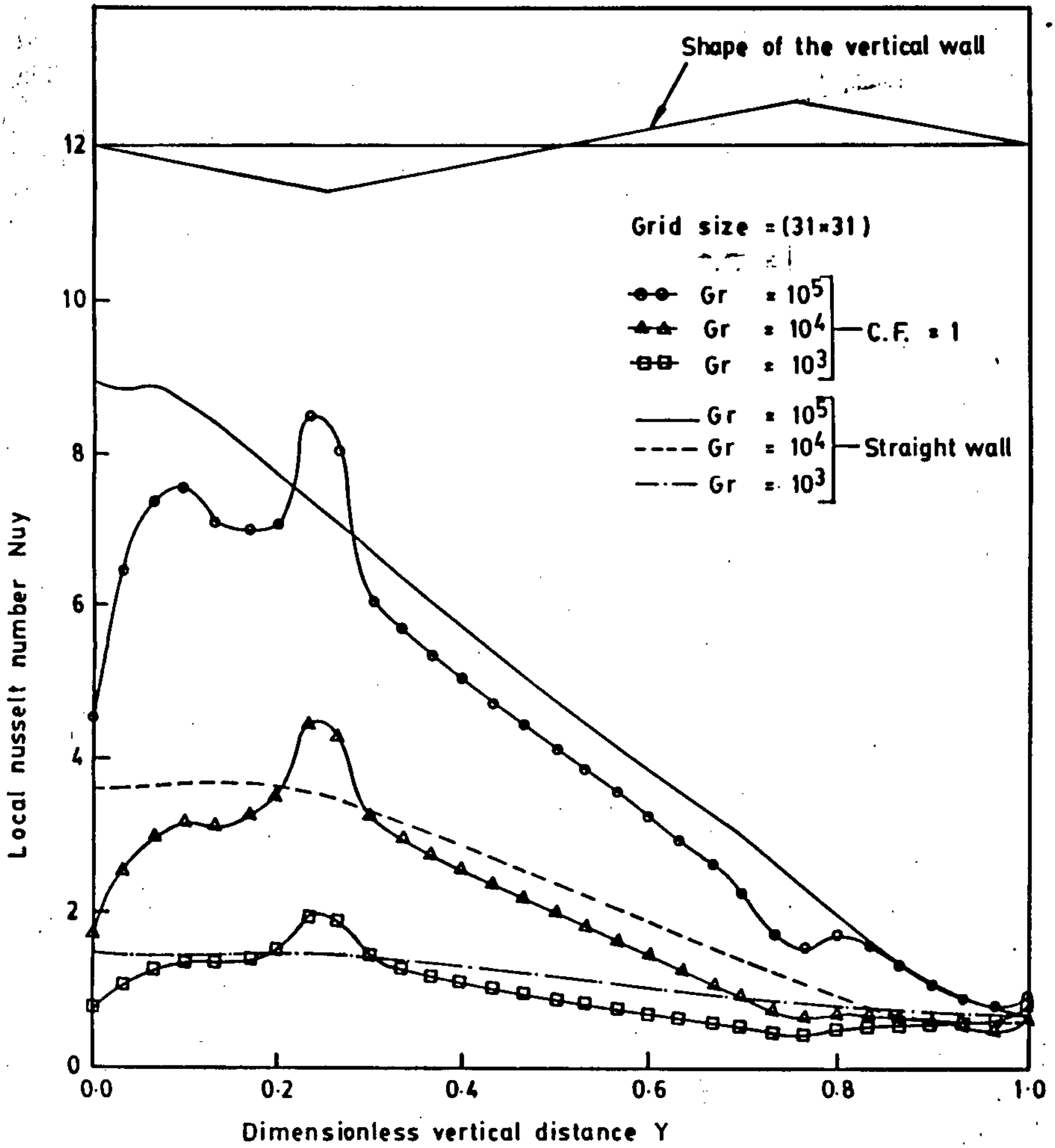


Fig. 11 Local Nusselt Number Distribution on the Hot Wall.

C.F. = 1 & Straight Wall.

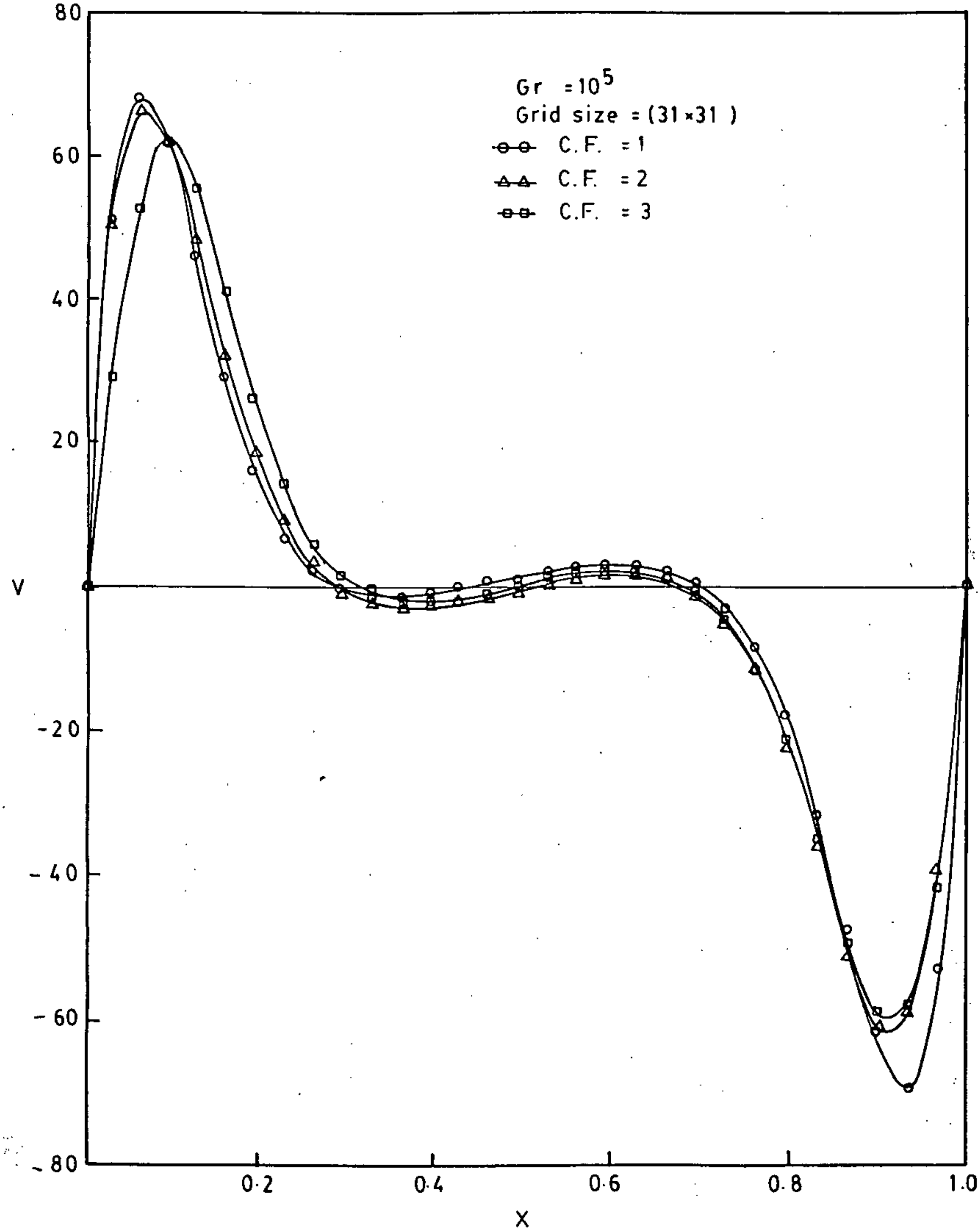


Fig. 12 Vertical Velocity Distribution at the Horizontal Mid Plane for Different Corrugation Frequency (C. F.).

Gr = 10^5.

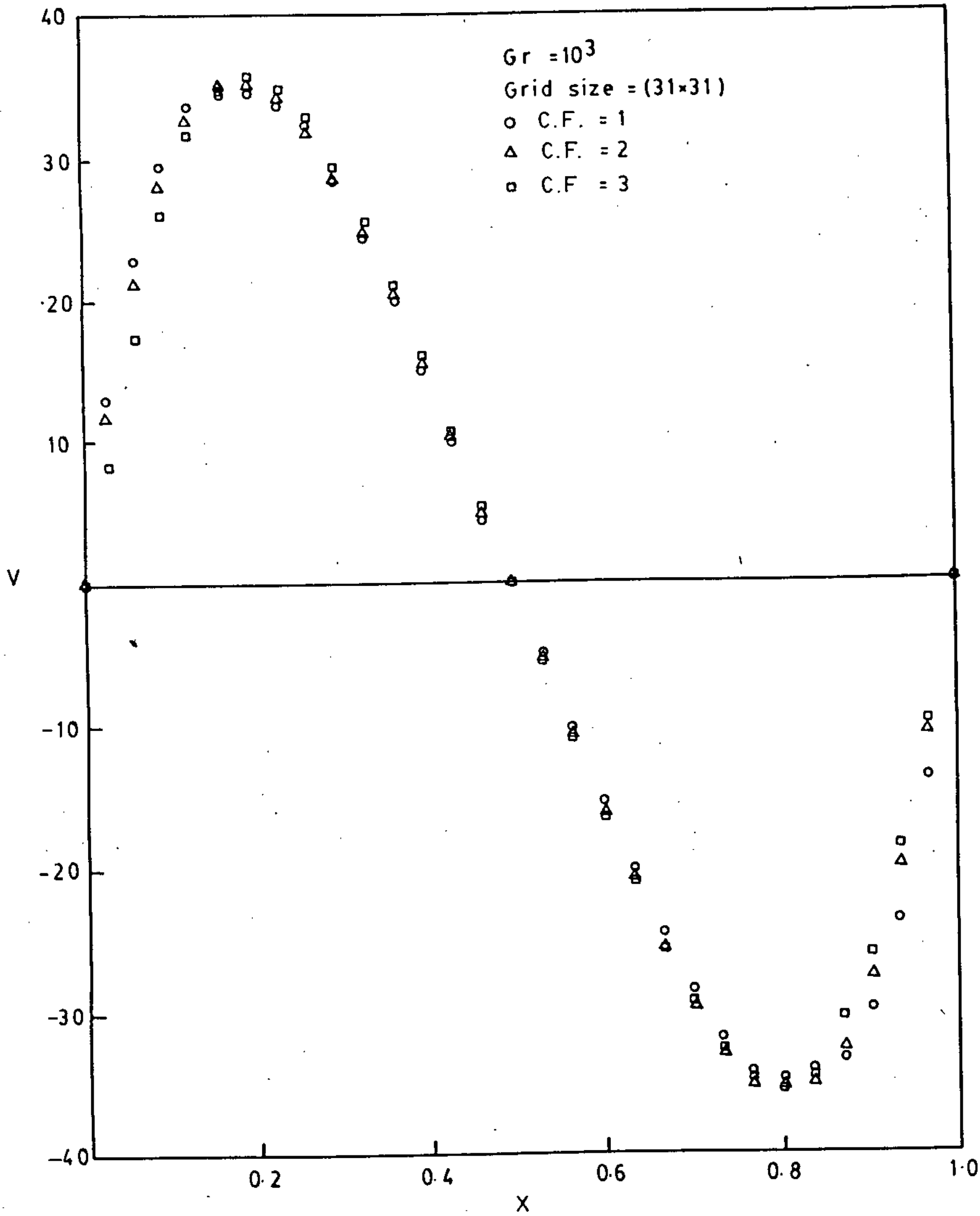


Fig. 13 Vertical Velocity Distribution at the Horizontal Mid Plane for Dofferent Corrugation Frequency (C.F.).

Gr = 10^3.

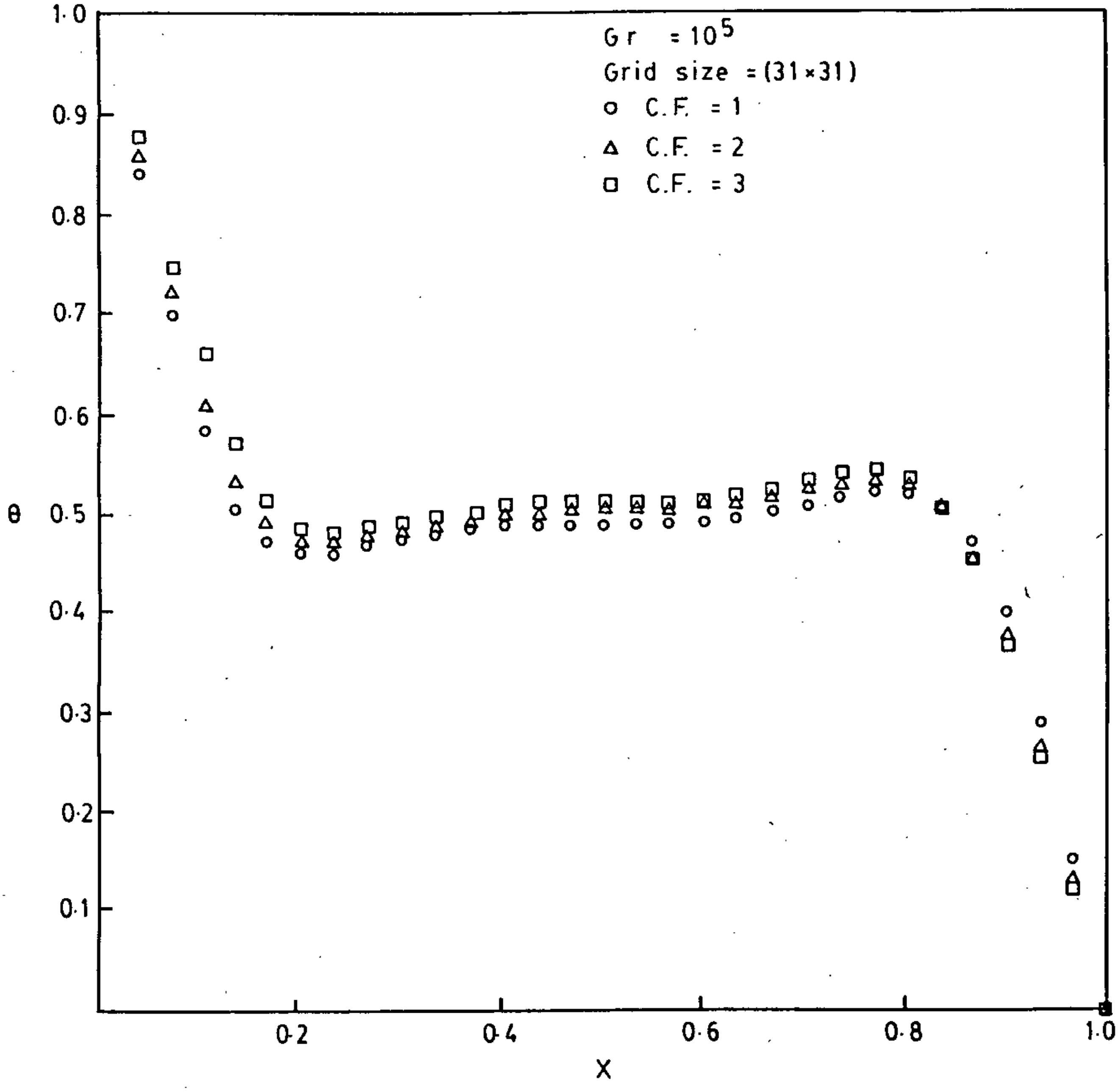


Fig. 14 Temperature Distribution at the Horizontal Mid Plane for Different Corrugation Frequency (C.F.).
Gr = 10^5.

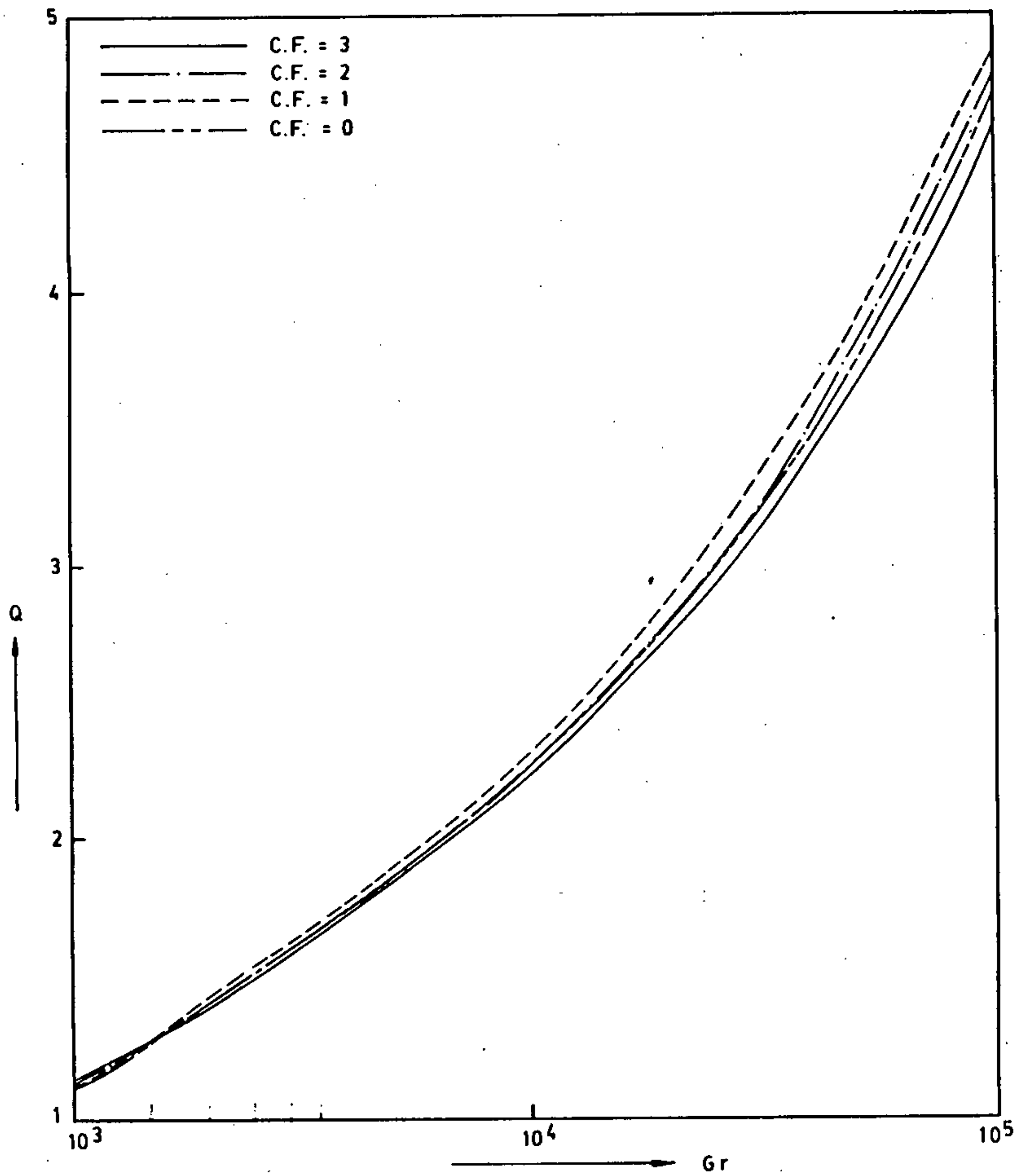


Fig. 15 Total Heat Flux Vs. Grashof Number (Gr).

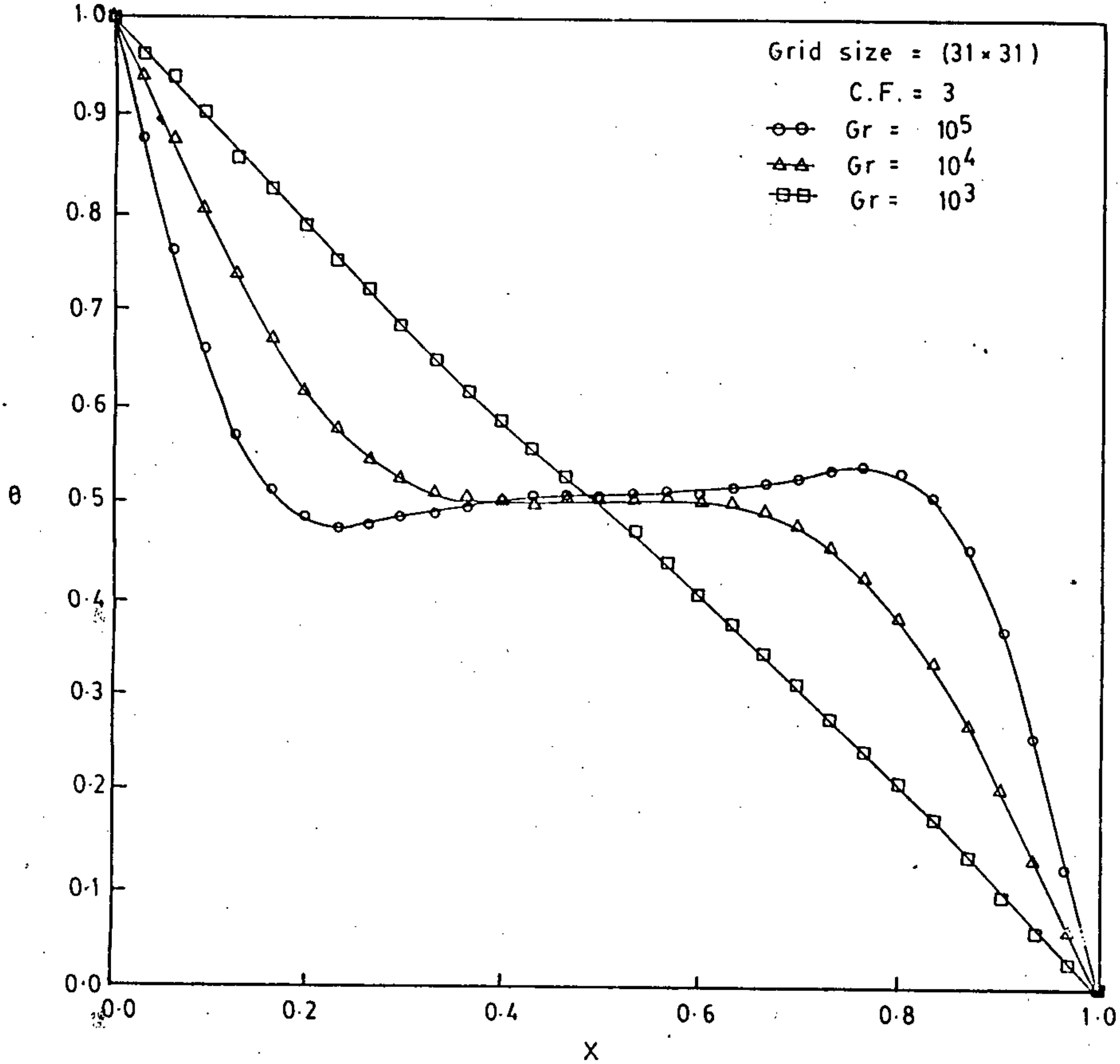


Fig. 16 Temperature Distribution at the Horizontal Mid Plane for Different Grashof Number (Gr).

C. F. = 3.

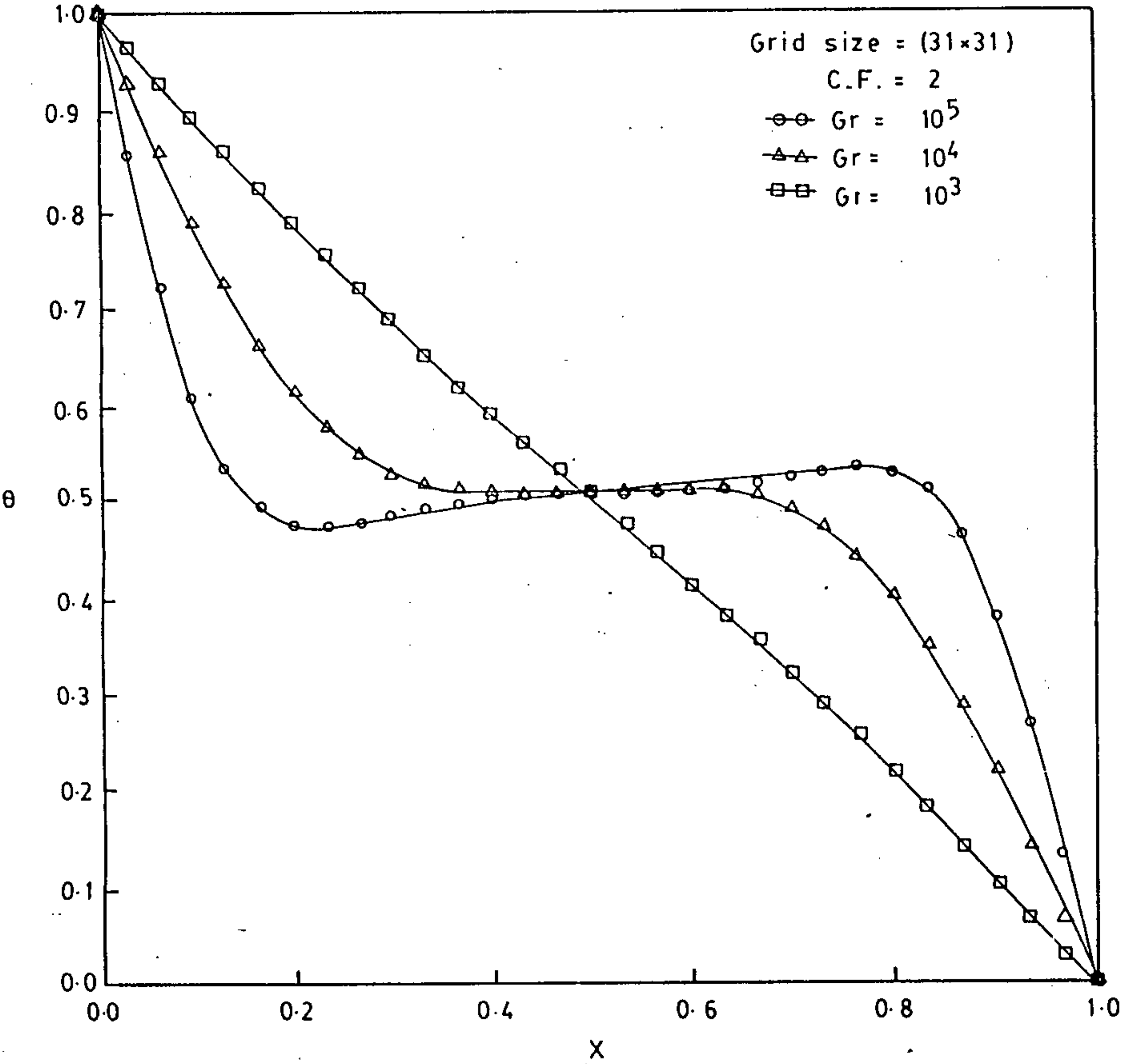


Fig.17 Temperature Distribution at the Horizontal Mid Plane for Different Grashof Number (Gr).

C.F. = 2.

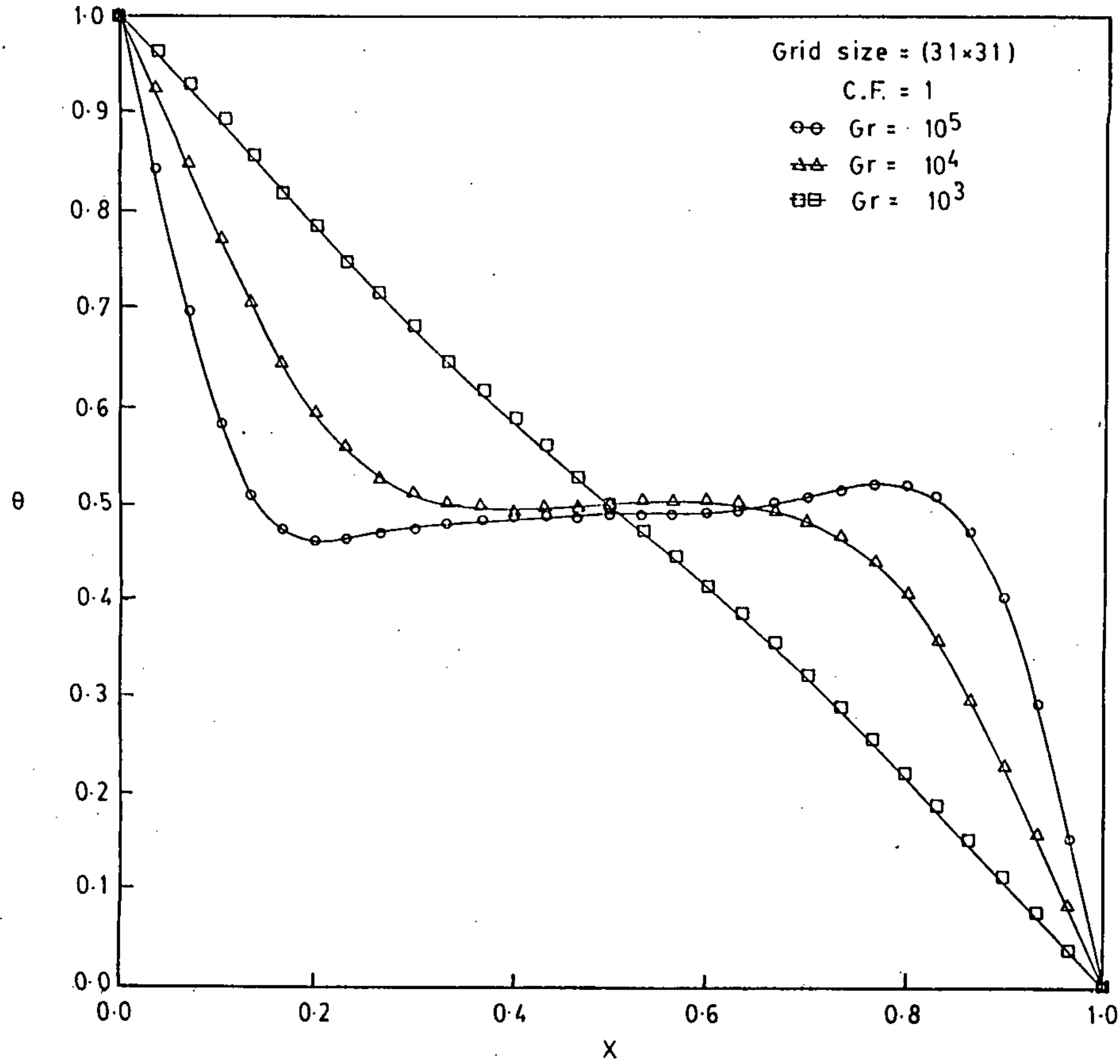


Fig. 18 Temperature Distribution at the Horizontal Mid Plane for Different Gr.

C.F. = 1.

APPENDIX - B

I. DISCRETIZATION OF THE GENERAL TRANSPORT EQUATION

A. The Fundamental Transport Equation.

The basic equation describing the transport of a conserved variable may be stated in a general form as

$$\vec{\nabla}\cdot(\rho \vartheta \phi) = \vec{\nabla}\cdot(\Gamma \vec{\nabla} \phi) + S \qquad \text{(B-1)}$$

where ϕ is the intensive property(property per unit mass) undergoing transport by a fluid of density and possessing a velocity vector field $\vec{v}$. Here, $\vec{\nabla}$ is the gradient operator defined as

$$\vec{\nabla} \equiv \hat{e}_x \frac{\partial}{\partial x} + \hat{e}_y \frac{\partial}{\partial y}$$

By applying the Gauss Divergence theorem to equation (B-1) the following integral formulation may be obtained:

$$\iint_{\sigma} (\rho \vec{\vartheta} \phi - \Gamma \vec{\nabla} \phi).\, \hat{n}\, d\sigma = \iiint_{\forall} S \, d\forall \qquad \text{(B-2)}$$

where $\forall$ is the volume enclosed by the surface σ and $\hat{n}$ is the outward unit normal vector at σ .

By rewriting

$$\vec{J} \equiv \rho \vec{v} \phi - \Gamma \vec{\nabla} \phi \tag{B-3}$$

equation (B-2) becomes

$$\iint_{\sigma} \vec{J} \cdot \hat{n} \, d\sigma = \iiint_{\forall} S \, d\forall \tag{B-4}$$

Thus, the term $\vec{J}$ represents the total flux vector due to convection and diffusion of ϕ .

B. **The Shape Function**

Consider an element with nodal velocities $\vec{v}_1$, $\vec{v}_2$, and $\vec{v}_3$ in terms of a global (fixed) coordinate system (x,y). Upon taking the average as follows,

$$\vec{v}_{avg} = (\vec{v}_1 + \vec{v}_2 + \vec{v}_3)/3 \tag{B-5}$$

with the components given by

$$u_{avg} = (u_1 + u_2 + u_3)/3 \tag{B-6}$$

$$v_{avg} = (v_1 + v_2 + v_3)/3 \tag{B-7}$$

a new elemental (local) coordinate system (X,Y) may be established with the X-direction chosen along the unit vector of $\vec{V}_{avg}$ in equation (B-5). Fig.(B-1) is the result of such an operation.

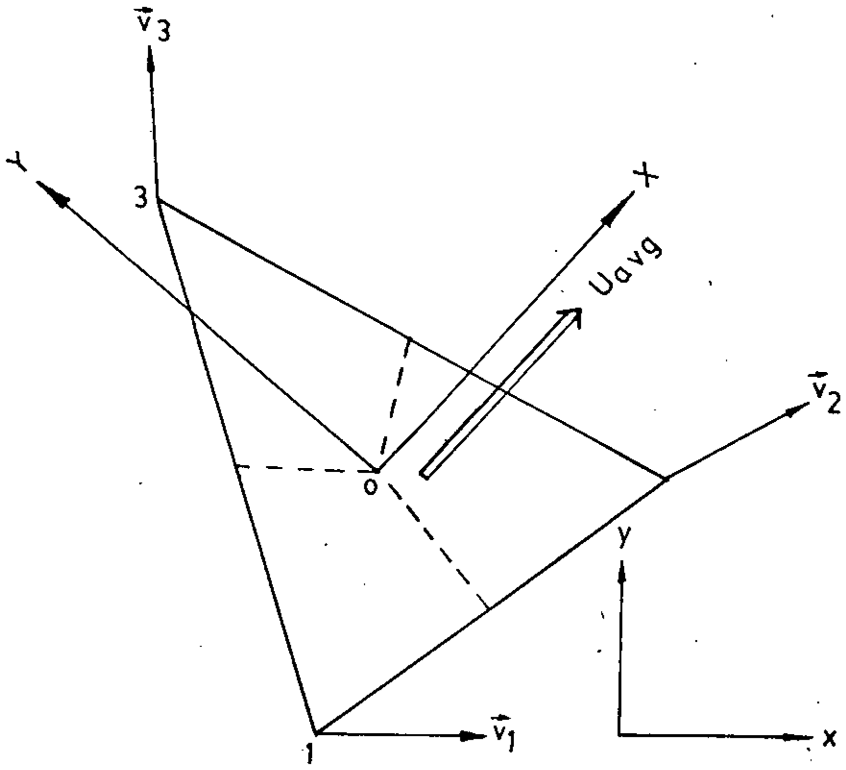


Fig. (B-1) Element with local coordinates defined in terms of nodal velocities.

The origin of this new coordinate system is fixed at the centroid of the element o. It is easily seen that with (x_o, y_o) as the origin of the global coordinates, the transformations between (X,Y) and (x,y) are given by the following equations:

$$U_{avg} = \sqrt{u_{avg}^2 + v_{avg}^2} \qquad \text{(B-8)}$$

$$\cos\theta = u_{avg}/U_{avg} \tag{B-9}$$

$$\sin\theta = v_{avg}/U_{avg} \tag{B-10}$$

$$X = (x - x_0)\cos\theta + (y - y_0)\sin\theta \tag{B-11}$$

$$Y = -(x - x_0)\sin\theta + (y - y_0)\cos\theta \tag{B-12}$$

$$U = u\cos\theta + v\sin\theta \tag{B-13}$$

$$V = -u\sin\theta + v\cos\theta \tag{B-14}$$

Based on the work of Baliga and Patankar [23] the shape function for ϕ in the (X,Y) coordinate system is given by

$$\phi = A\exp\left(\frac{\rho U_{avg}}{\Gamma} X\right) + BY + C \tag{B-15}$$

where A, B, and C are the parameters to be determined from the constraints

$$\phi = \phi_i \quad \text{at } X = X_i,\ Y = Y_i \quad i = 1 \ldots 3 \tag{B-16}$$

This choice for the shape function is made for the following reasons:

1) The exponential term allows upstream weighting of ϕ .

Specifically, ϕ at any point in the element is strongly dependent on the values of the upstream points when U_{avg} is sufficiently large. This is a very desirable feature because the exponential function in equation (B-15) describes the exact solution of the convection-diffusion problem in one dimension without source terms.

2) Equation (B-15) is based on the local coordinates, which are aligned with the average flow velocity within the element. Therefore, this shape function accounts for the two dimensionality of the flow field and thus reduces false (numerical) diffusion considerably.

3) Let Peclet number Pe_{Δ} be defined as

$$Pe_{\Delta} \equiv \rho U_{avg} \Delta X / \Gamma$$

where ΔX is a characteristic element dimension. Then, as Pe_{Δ} approaches zero, the shape function (B-15) reduces to a linear form in (X,Y) or (x,y). This type of function is commonly used in the finite element method.

The determination of the constants A, B, and C now follows. Let the following definitions be made:

$$X_{max} \equiv \text{largest of } X_1, X_2, X_3 \qquad \text{(B-17)}$$

$$X_{min} \equiv \text{smallest of } X_1, X_2, X_3 \tag{B-18}$$

and

$$Z \equiv \frac{\Gamma}{\rho U_{avg}} \left\{ \exp\left[\frac{Pe_\Delta (X - X_{max})}{X_{max} - X_{min}}\right] - 1 \right\} \tag{B-19}$$

Now, based on equation.(B-19), equation (B-15) may be rewritten as

$$\phi = AZ + BY + C \tag{B-20}$$

and based on the requirement of (B-16) it follows that

$$A = L_i \phi_i \tag{B-21}$$

$$B = M_i \phi_i \tag{B-22}$$

$$C = N_i \phi_i \tag{B-23}$$

where repeated subscripts imply summation i = 1....3. In particular,

$$\begin{aligned} L_1 &= (Y_2 - Y_3)/DET \\ L_2 &= (Y_3 - Y_1)/DET \\ L_3 &= (Y_1 - Y_2)/DET \end{aligned} \tag{B-24}$$

$$\begin{aligned} M_1 &= (Z_3 - Z_2)/DET \\ M_2 &= (Z_1 - Z_3)/DET \\ M_3 &= (Z_2 - Z_1)/DET \end{aligned} \tag{B-25}$$

and

$$N_1 = (Z_2Y_3 - Z_3Y_2)/DET$$

$$N_2 = (Z_3Y_1 - Z_1Y_3)/DET \qquad (B-26)$$

$$N_3 = (Z_1Y_2 - Z_2Y_1)/DET$$

where DET is the determinant

$$DET = Z_1(Y_2 - Y_3)+Z_2(Y_3 - Y_1)+Z_3(Y_1 - Y_2) \qquad (B-27)$$

Upon substituting equations(B-21,B-22,and B-23) into (B-20),the result is

$$\phi = F_i\phi_i \qquad (B-28)$$

where $F_i = L_iZ + M_iY + N_i$ are the shape functions.

C. Flux Calculation

Equation (B-3) may be written in terms of its components as

$$J_x = \rho U\phi - \Gamma \frac{\partial \phi}{\partial X} \qquad (B-29)$$

$$J_y = \rho V\phi - \Gamma \frac{\partial \phi}{\partial Y} \qquad (B-30)$$

Using equations (B-19) through (B-23), (B-29) and (B-30) may be rewritten as

$$J_x = (\rho f_i - \Gamma L_i)\ \phi_i \tag{B-31}$$

$$J_y = (\rho g_i - \Gamma M_i)\ \phi_i \tag{B-32}$$

where

$$f_i \equiv (U-U_{avg})L_iZ + U(M_iY+N_i) \qquad i = 1...3 \tag{B-33}$$

$$g_i \equiv V(L_iZ + M_iY + N_i) \qquad i = 1....3 \tag{B-34}$$

The expressions for the flux vector components are now available. The integral formulation of the basic equation (B-4) involves flux calculations across each control volume boundary as per its left hand term. On viewing Fig. 3 (See APPENDIX - A), it is evident that the internal control volume surrounding node P possesses a surface that is composed of links, pairs of which belong to elements sharing the node. Moreover, each element is made up of three portions of three distinct control volumes. It is therefore convenient to visit each element and calculate the fluxes across its three links and follow up with an assembly process. The assembly process essentially involves taking into account in a systematic manner the elemental flux contributions to the associated control volume portions. Once all the elements in the domain are visited and their contributions assembled, the flux calculations for all the control volumes are complete. The particulars for the elemental flux contributions now follows.

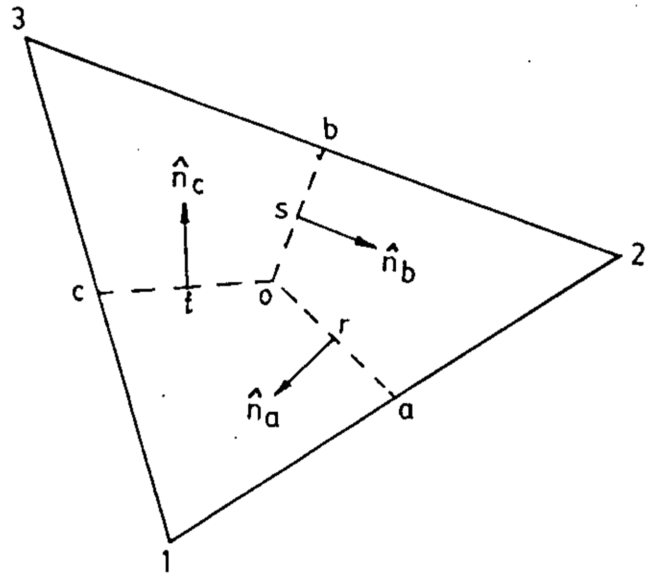


Fig. (B-2) Element with quadrature points and link unit normal vectors.

A typical element is shown in Fig.(B-2) in detail. It may be noted that the vertices of the element are numbered in a counterclockwise fashion. If all elements follow the same local numbering convention, the flux calculation scheme that is to be presented applies without any modifications. The claculation of fluxes of the general variable ϕ across each link is performed by means of Simpson's quadrature rule. The integration points are a,b,c,r,s,t, and o, as shown. Thus, it is clear that the three links within the element contain the groups a-r-o, b-s-o, and c-t-o. Moreover, the arrows drawn on each link denote the corresponding normal unit vectors. Upon defining the following radius vectors

$$\vec{r}_{oa} \equiv X_a\hat{I} + Y_a\hat{J} \qquad \text{(B-35)}$$

$$\vec{r}_{ob} \equiv X_b\hat{I} + Y_b\hat{J} \qquad \text{(B-36)}$$

$$\vec{r}_{oc} \equiv X_c\hat{I} + Y_c\hat{J} \qquad \text{(B-37)}$$

it follows that

$$\hat{n}_a = \frac{Y_a\hat{I} - X_a\hat{J}}{|\ \vec{oa}\ |} \qquad \text{(B-38)}$$

$$\hat{n}_b = \frac{Y_b\hat{I} - X_b\hat{J}}{|\ \vec{ob}\ |} \qquad \text{(B-39)}$$

$$\hat{n}_c = \frac{Y_c\hat{I} - X_c\hat{J}}{|\ \vec{oc}\ |} \qquad \text{(B-40)}$$

where

$\hat{n}_a \equiv$ normal unit vector to link oa

$\hat{n}_b \equiv$ normal unit vector to link ob

$\hat{n}_c \equiv$ normal unit vector to link oc

With the link normal unit vectors established, the fluxes are as follows:

$$\int_{link\ oa} \vec{J}\cdot\hat{n}\,d\sigma = \int_0^{|\vec{oa}|}\left[J_x Y_a - J_Y X_a\right] dl \qquad \text{(B-41)}$$

$$\int_{link\ ob} \vec{J}\cdot\hat{n}\,d\sigma = \int_0^{|\vec{ob}|}\left[J_x Y_b - J_Y X_b\right] dl \qquad \text{(B-42)}$$

$$\int_{link\ oc} \vec{J}\cdot\hat{n}\,d\sigma = \int_0^{|\vec{oc}|}\left[J_x Y_c - J_Y X_c\right] dl \qquad \text{(B-43)}$$

The application of Simpson's rule to equations (B-41) through (B-43) yields

$$\int_{link\ oa} \vec{J}.\hat{n}\,d\sigma = [(J_X^a + 4J_X^\gamma + J_X^o)Y_a - (J_Y^a + 4J_Y^\gamma + J_Y^o)X_a]/6 \qquad \text{(B-44)}$$

$$\int_{link\ ob} \vec{J}.\hat{n}\,d\sigma = [(J_X^b + 4J_X^\beta + J_X^o)Y_b - (J_Y^b + 4J_Y^\beta + J_Y^o)X_b]/6 \qquad \text{(B-45)}$$

$$\int_{link\ oc} \vec{J}.\hat{n}\,d\sigma = [(J_X^c + 4J_X^\xi + J_X^o)Y_c - (J_Y^c + 4J_Y^\xi + J_Y^o)X_c]/6 \qquad \text{(B-46)}$$

Substituting equations (B-31) and (B-32) into (B-44) and simplifying, the flux across link oa may be expressed (with repeated subscripts implying summation) as,

$$\int_{link\ oa} \vec{J}\cdot\hat{n}\,d\sigma = \lambda_i^{oa}\phi_i \qquad \text{(B-47)}$$

where

$$\lambda_i^{oa} = \frac{\rho}{6}[(f_i^a + 4f_i^\gamma + f_i^o)Y_a - (g_i^a + 4g_i^\gamma + g_i^o)X_a] - \Gamma[L_i Y_a - M_i X_a] \qquad \text{(B-48)}$$

It may be pointed out that in equation (B-48) the subscripts a,r, and o imply the evaluation of equations (B-31) and (B-32) at the corresponding locations in conjunction with (B-33) and (B-34). The relations for the links ob and oc can be deduced analogously and stated. Thus, for link ob,

$$\int_{link\ ob} \vec{J} \cdot \hat{n}\, d\sigma = \lambda_i^{ob} \phi_i \tag{B-49}$$

with

$$\lambda_i^{ob} = \frac{\rho}{6}[(f_i^b + 4f_i^s + f_i^o)Y_b - (g_i^b + 4g_i^s + g_i^o)X_b] - \Gamma[L_i Y_b - M_i X_b] \tag{B-50}$$

and for link oc,

$$\int_{link\ oc} \vec{J} \cdot \hat{n}\, d\sigma = \lambda_i^{oc} \phi_i \tag{B-51}$$

with

$$\lambda_i^{oc} = \frac{\rho}{6}[(f_i^c + 4f_i^t + f_i^o)Y_c - (g_i^c + 4g_i^t + g_i^o)X_c] - \Gamma[L_i Y_c - M_i X_c] \tag{B-52}$$

With the formulations for the flux across the links available, the elemental contributions to its control volume portions are constructed next. This leads to an element flux matrix which facilitates the assembly process. This is described next.

Attention is again drawn to Fig. (B-2). Let the control volume segment containing node i(i=1...3) be referred to as CV_i. For example, it may be observed that the elemental flux contribution to CV_1 is effected via links oa and oc. Moreover, the unit normal for oa points "into" CV_1 whereas that for link oc points "out" from same. This is so due to the choice of a right-handed coordinate system and the resulting vector products given in equations (B-38) through (B-40). With these points in mind, it is easy to see that the net efflux Ξ of ϕ from CV_1 via links oa and oc may be written as

$$\Xi_1 = (\lambda_i^{oc} - \lambda_i^{oa})\,\phi_i \tag{B-53}$$

In a similar manner it follows that the net effluxes Ξ_2 and Ξ_3 concerning CV_2 and CV_3, respectively, are expressed as

$$\Xi_2 = (\lambda_i^{oa} - \lambda_i^{ob})\,\phi_i \tag{B-54}$$

$$\Xi_3 = (\lambda_i^{ob} - \lambda_i^{oc})\,\phi_i \tag{B-55}$$

Upon making the following definitions,

$$A_{1i} \equiv \lambda_i^{oc} - \lambda_i^{oa} \tag{B-55a}$$

$$A_{2i} \equiv \lambda_i^{oa} - \lambda_i^{ob} \tag{B-55b}$$

$$A_{3i} \equiv \lambda_i^{ob} - \lambda_i^{oc} \tag{B-55c}$$

Equations (B-53), (B-54), and (B-55) may be recast in matrix form as

$$\begin{Bmatrix} \Xi_1 \\ \Xi_2 \\ \Xi_3 \end{Bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{Bmatrix} \tag{B-56}$$

The 3 X 3 matrix above is called the Element Flux Matrix (EFM). For reasons to be explained later, certain modifications to the coefficients A_{11}, A_{22} and A_{33} are made in what follows.

It is seen that the equation for the conservation of mass is obtained through equation (B-2) when ϕ is set to unity. Further, let the mass flow rates out of CV_1, CV_2, and CV_3 via their associated link pairs be denoted by Π_1, Π_2 and Π_3, respectively. It therefore follows immediately that

$$\Pi_i \phi_i = (A_{i1} + A_{i2} + A_{i3})\phi_i, \qquad i = 1 \ldots 3$$

If the above indical equations are subtracted from the equation set (B-56) the following result is obtained:

$$\begin{Bmatrix} \Xi_1 - \Pi_1\phi_1 \\ \Xi_2 - \Pi_2\phi_2 \\ \Xi_3 - \Pi_3\phi_3 \end{Bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{Bmatrix} \tag{B-57}$$

where the diagonal coefficients of the EFM are redefined as,

$$A_{11} \equiv -(A_{12} + A_{13})$$

$$A_{22} \equiv -(A_{21} + A_{23})$$

$$A_{33} \equiv -(A_{31} + A_{32})$$

Even though the expressions for the fluxes are now changed, these will not affect the final solution when convergence is reached because the velocity fields will obey continuity which, in turn will cause the assembled values of II_i to vanish. Furthermore, this feature of the EFM will result in the point coefficient in the nodal equation for ϕ to equal magnitude of the sum of its corresponding neighbor coefficients, which is an important requirement for iterative stability as discussed by Patankar [19].

Attention may now be drawn to the right hand term of equation (B-4) which is a volume integral. Since only 2-dimensional problems are being considered, it is strictly an area integral. As per the assertion made earlier that the source term S in (B-4) is constant over any particular element, this integral becomes

$$\int_{A^e} S \, d(A^e) = S^e A^e \qquad \text{(B-58)}$$

where A^e denotes the area of the element under consideration. The structure of the links within a triangular element assure that each control volume portion has an area equal to one-third of the total elemental area. Therefore, the contribution of the integral in equation (B-58) to each portion is given in terms of an Element Load Vector (ELV):

$$\{ELV\} = \frac{S^e A^e}{3} \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \qquad \text{(B-59)}$$

Based on the setup of both the EFM and ELV, and with respect to equation (B-4), it is possible to combine the two to yield the elemental conservation equation in a compact form:

$$[EFM] \{ \Phi \} = \{ELV\} \qquad \text{(B-60)}$$

where

$$\{ \Phi \} \equiv \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{Bmatrix}$$

It is emphasized here that equation (B-60) is only a partial set which when combined in an appropriate manner via the assembly process with those of other elements yields the complete control volume conservation equations that are to be solved for. This assembly procedure is detailed in the following section.

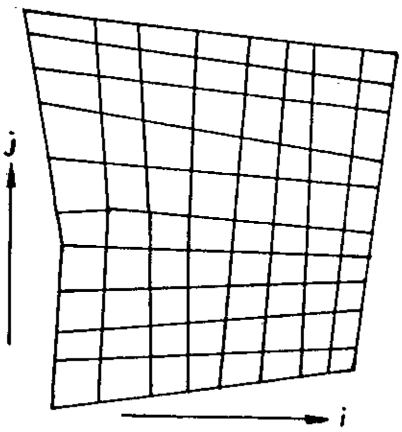
D. The Assembly

The solution of fluid flow problems by numerical methods essentially involves the solution of a system of nominally linear equations which may be stated in matrix form as

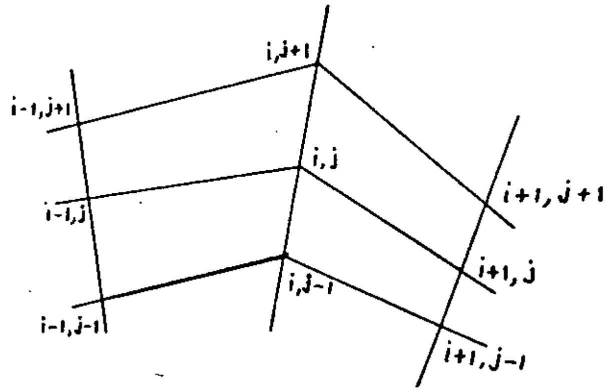
$$\begin{bmatrix} c_{11} & c_{12} & \cdots\cdots & c_{1n} \\ c_{21} & c_{22} & \cdots\cdots & c_{2n} \\ \cdot & \cdot & \ddots & \cdot \\ \cdot & \cdot & & \cdot \\ c_{n1} & c_{n2} & \cdots\cdots & c_{nn} \end{bmatrix} \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \cdot \\ \cdot \\ \phi_n \end{Bmatrix} = \begin{Bmatrix} r_1 \\ r_2 \\ \cdot \\ \cdot \\ r_n \end{Bmatrix} \tag{B-61}$$

where $[c]_{nxn}$ is known as the Global Stiffness Matrix (GSM), $\{r\}_{nx1}$ the Global Load Vector (GLV), and n is the number of nodes in the domain. Frequently, the GSM is a sparse matrix, that is, many of its components are zero. The extent of sparseness depends largely on the domain discretization scheme and the manner in which the equations are formulated. If the GSM is sufficiently sparse, the form of storage as shown above becomes wasteful and an alternative substructuring becomes more desirable. This substructuring technique leads to a considerable reduction in storage requirements at the cost of limiting the freedom of domain discretization. For the problems considered, a rectangular domain discretization is adopted which is described next.

An example of a rectangular domain discretization is depicted in Fig. (B-3a). The domain itself need not be of a

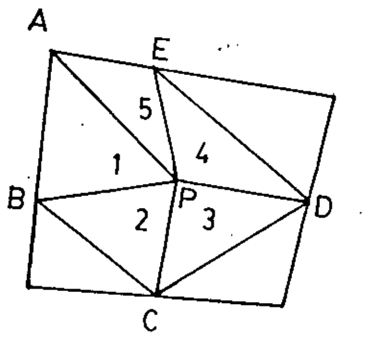


(a)

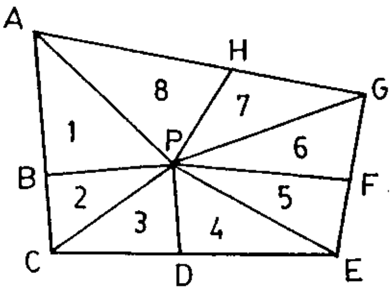


(b)

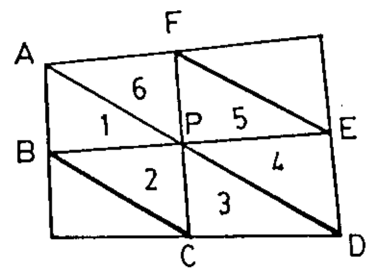
Fig. (B-3) Rectangular domain discretization.



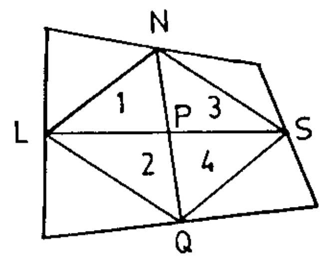
(a)



(b)



(c)



(d)

Fig. (B-4) Some examples of triangulation of quadrilaterals.

rectangular shape. It may be observed that the region is subdivided into quadrilaterals by a family of horizontal and vertical line segments. Fig. (B-3b) focuses on a subregion of the domain in detail where the horizontal line segments are denoted by the index j and the vertical by the index i. A typical node in the domain is then identified by the index pair (i,j). Its right hand neighbor is specified by (i+1,j),etc.

Next, the elements are obtained by subdividing each quadrilateral by a diagonal. Some examples of this triangulation are shown in Fig. (B-4). It is clear that node P has at most eight neighbors as exemplified in Fig. (B-4b), which are A,B,C,D,E,F,G, and H. This is a direct consequence of the rectangular discretization process. Because of this condition, a Nodal Equation Matrix, (NEM) may be defined as follows:

$$AP(i,j,m) \qquad i,j=1...3$$

where m denotes the global node number and i, j the locations of its neighbors. This matrix is used to form the coefficients of the equations for each node in the domain instead of the sparse GSM. For example, AP(2,2,m) is the coefficient of the node m (i,j) itself while AP(3,1,m) denotes the neighbor coefficient at the top left (i-1,j+1) of the nodal equation for m. Prior to the assembly, both the GSM and GLV are initialized to zero. To proceed further with details, let us consider Fig. (B-4d) as an example.

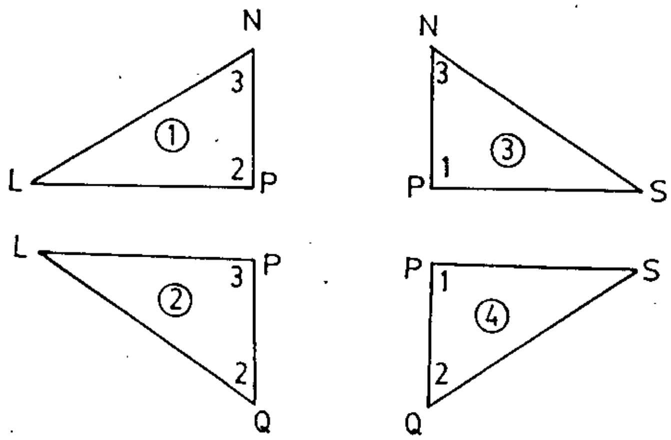


Fig. (B-5) Element breakup of Fig.(B-4d).

Taking guidance from Fig. (B-4d), the element group forming the control volume around node P is fragmented and shown in Fig. (B-5). It may be observed here that the elements (1),(2),(3) and (4) possess local node numbers that correspond to the global ones as shown in Table (B-1). Suppose that the EFM and ELV for element (1) are available. With reference to Table (B-1), it is seen that the element flux contribution to CV_1 (portion of control volume around node L) is given by

$$A_{11}\phi_L + A_{12}\phi_P + A_{13}\phi_K$$

Similarly, for CV_2 (portion around P) the flux contribution is

$$A_{21}\phi_L + A_{22}\phi_P + A_{23}\phi_K$$

while that for CV_3 (portion surrounding K) is

$$A_{31}\phi_L + A_{32}\phi_P + A_{33}\phi_K$$

Table (B-1). Local-global node correspondence for Fig. (B-4d).

ELEMENT	LOCAL NODE	GLOBAL NODE
	1	L
1	2	P
	3	N
	1	L
2	2	Q
	3	P
	1	P
3	2	S
	3	N
	1	P
4	2	Q
	3	S

In a like manner, the ELV contributions of elements (1) are added on to the GLV at r_L, r_P, and r_K. Thus, the elemental flux and load contributions for (1) may be summarized in Table (B-2). The other elements (2),(3) and (4) are handled in exactly the same

Table (B-2). Element matrix and load vector assembly.

Element Coefficient	⟶ NEM Coefficient
A_{11}	AP(2,2,L)
A_{12}	AP(2,3,L)
A_{13}	AP(3,3,L)
A_{21}	AP(2,1,P)
A_{22}	AP(2,2,P)
A_{23}	AP(3,2,P)
A_{31}	AP(1,1,N)
A_{32}	AP(1,2,N)
A_{33}	AP(2,2,N)
ELV_1	r_L
ELV_2	r_P
ELV_3	r_N

manner. This procedure then yields the complete coefficients for the nodal equation concerning P

$$AP(i,j,P) \quad i,j = 1...3$$

and the corresponding global load component r_P. It is therefore clear that by visiting all the elements, the NEM and GLV may be constructed completely. For the sake of convenience, the nodal equation for an arbitrary point p may be expressed as

$$a_p \phi_p + \sum_{nb} a_{nb} \phi_{nb} = r_p \qquad \text{(B-62)}$$

where (nb) indicates the summation over the neighbor nodes of p, a are the corresponding coefficients obtained from the NEM, and the term r_p denotes the effective source term or the global load component.

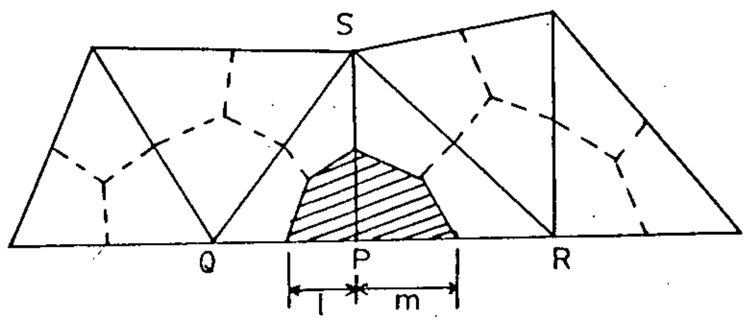


Fig. (B-6) A typical boundary control volume shown shaded.

E. Boundary Conditions

When all the elements have been visited and assembled, the resulting equations are immediately available for solution "only" for the internal nodes. This is because all interior nodes are surrounded by complete control volumes. On the other hand, all boundary nodes are enclosed by incomplete or "half" control volumes as shown in Fig. (B-6), where it is seen that they are bounded at the bottom by links consisting of element sides. Based on what has been discussed, it is clear that the flux through these boundary surfaces have yet to be accounted for. These boundary conditions may be categorized into three classes:

1) Specified ϕ boundary: Here, the nodal equation (B-62) is replaced by one with neighbor coefficients set to zero, the point coefficient to unity, and the global load component to the specified value. The replaced equation coefficients may be stored elsewhere and retrieved later to calculate the flux across the control volume boundary. For example, the heat transfer and shear stresses at the boundaries may be obtained where velocities and temperatures, respectively, are specified.

2) Specified diffusion boundary: In this case, the specified diffusion efflux is integrated across the boundary links by the trapezoidal rule and appropriately appended to the partially assembled nodal equation. As an example, let us consider the boundary node p and its associated control volume (shaded). Further, let the integrated diffusion efflux of ϕ at the boundary links l and m be prescribed as $\Xi_p = h(\phi_p - \phi_\infty)$ where ϕ_∞ is some reference value. Then,

the available partial nodal equation for the node p is modified as follows:

$$(h + a_p)\phi_p + \sum_{nb} a_{nb}\,\phi_{nb} = r_p + h\,\phi_\infty \qquad \text{(B-63)}$$

It may be noted that without any modifications, the boundary nodal equations default to a flux free boundary condition. This is precisely the case when symmetry boundaries (channel centerline) or, in the case of the energy transport, insulated walls are encountered.

3) **Exit Boundary :** This type of boundary is present where flow is leaving the domain. Since there is no knowledge of conditions downstream of this boundary, it is assigned a convection-only condition. Referring to Fig. (B-6) again, the efflux across the control volume boundary links l and m are given by $-\dot{m}_p\phi_p$ where $\dot{m}_p$ is the mass flow rate into the domain. It is to be recalled now that the Element Flux Matrix for ϕ was obtained in slightly modified form. To maintain this consistency one would subtract from the left side of the available nodal equation at node p the term $-\dot{m}_p\phi_p$. The convection-only boundary condition would next be implemented by adding to the right side of the resulting equation the term $-\dot{m}_p\phi_p$, which represents the influx of ϕ through the boundary links l and m. The net result is clearly a do-

nothing process. It is therefore obvious that with the procedure described for obtaining the EFM and the nature of the assembly process, no modifications to the exit boundary nodal equations are needed.

Once the boundary condition modifications are complete, the nodal equation set is ready for solution. In a fluid flow problem where several of such sets, each representing different variables (e.g. velocity, enthalpy) undergo solution an iterative approach is preferred. This is because the velocity fields, which play a key role in calculating the equation coefficients themselves are not known. Basically, each field variable distribution is solved for in turn until the coefficients of the relevant nodal equations cease varying beyond a certain tolerance. The overall procedure may be outlined as follows:

1) Guess the distribution of the various ϕ in the domain, such as velocity, enthalpy, pressure,etc.
2) Obtain nodal equations and apply necessary boundary conditions.
3) Solve these equations and check for convergence. If convergence has been reached, stop computation. If values still changing, go back to step (2) with the currently available values of ϕ.

It may be mentioned here that the actual solution of any particular nodal equation set during the overall iteration process (step (3) above) need not be carried to extreme accuracy because the coefficients themselves are temporary. Moreover, as the coefficients approach convergence, less effort is required to obtain solution of a set. It is for these reasons that an iterative technique such as the line-by-line tridiagonal matrix algorithm technique is adopted. Often, during the overall iteration process, the updating of ϕ needs to undergo relaxation to maintain stability. These details may be found in Patankar [19].

II. PROGRAM LISTING

COMVORT.BLK

```
   parameter(im=49,jm=49,nmx=im*jm,nlt=2*(im-1)*(jm-1),
#              mxb=2*(im+jm-2))
 COMMON /ARRAY/ VOR(nmx),PP(nmx),UFLX(nmx)
 COMMON /ARRA2/ RHP(nmx),SU(nmx),DU(nmx)
 COMMON /CONST/ GRAV,BLK,IMAXC,JMAXC,IPM,JPM,IPMM,JPMM,
1        IPMT,JPMT,IPMTM,JPMTM,JTL,JTR,IM1,JM1,IM2,JM2,NODT,
2        NODC,NELTC,NUMELC,NELTOT,UTOP,UBOT,VTOP,VBOT,ULEF,URIT,
3        VLEF,VRIT,RHO,HGT,VIS,CONST,NELEF,NERIT,NBOTOP,
4        IBOTOP,ILEFT,IRIGHT,NB1,NB2,NB3,NBMAX,THETA
 COMMON /DXDY/ DX(jm-1),DY(im-1),MAP(im+2,jm+2),
1              PHI(im+2,jm+2),UIN(im)
 COMMON /ELEM/ NCA(nlt,3),SUEL(nlt),SVEL(nlt),UOUT(im)
 COMMON /NODL/ GAM(2,nmx),SV(nmx),U(nmx),V(nmx),UH(nmx),
1        VH(nmx),DV(nmx),P(nmx),PSI(im,jm),
2        SMASS(nmx),VFLX(nmx)
 COMMON /GRID/ NP(im,jm),X(nmx),Y(nmx),RH(nmx)
 COMMON /MATRX/ A(3,3,nmx),AP(3,3,nmx),VOL(nmx),AREA3(nlt),
1        ALPHA(nlt,3),BETA(nlt,3),APSI(3,3,nmx)
 COMMON /BELE/ NBND(mxb)
 COMMON /MISC/ XT(3),YT(3),XN(3),YN(3),XTD(3),YTD(3),DCU(3),
1        DCV(3),ESM(3,3),EFMP(3,3),H(3)
 COMMON DUDY,BURG,EXPX,EXPY
 LOGICAL DUDY
 COMMON RE
 LOGICAL NCLRNC,UNIFRM,UV2,BV2,VCALC,PLOTY,BURG,READ,WRITE
```

```
      Program main
c
c       This is the CVFEM code that uses the Streamfunction
c       -vorticity formulation of the N.S. equations. This
c       program solves the test problem, "Natural Convection
c       Heat Transfer in a Square Duct with V-Corrugated
c       Vertical Walls". Relaxation factor for wall vorticity
c       of 0.5 and that for vorticity of 0.5 are recommended.
c       For temperature, also use 0.8.
c
c
      include'comvort.blk'
      nclrnc=.true.
      open(13,file='in.dat',status='unknown')
      read(13,*) unifrm,vcalc,uv2,bv2,burg,gr,relt,crit,dudy
      read(13,*) read,write
      read(13,*) ngr,ipm,jpm,maxit,bvrel,relv,re,ploty,expx,expy
      read(13,*) ampl, freq
      close(13)
      open(11,file='out.dat',status='unknown')
      nodc=ipm*jpm
      neltc=2*(ipm-1)*(jpm-1)
      numelc=neltc
      ipmp=ipm+1
      jpmp=jpm+1
      ihalf=ipmp/2
      jhalf=jpmp/2
      imaxc=ipm+2
      jmaxc=jpm+2
      ipmt=5
      jpmt=5
      jtl=4
      jtr=8
      ipmm=ipm-1
      jpmm=jpm-1
      ipmtm=ipmt-1
      jpmtm=jpmt-1
      rho=1.0
c
      im1=imaxc-1
      im2=imaxc-2
      jm1=jmaxc-1
      jm2=jmaxc-2
c ******* set double density grid at the boundaries *******
 7120 continue
      prody=expy
      ipend=ihalf-3
      do 7123 ip=1,ipend
 7123 prody=(prody+1.0)*expy
      dy(1)=0.5/(1.+prody)
      dy(ipmm)=dy(1)
```

```
      ie=ipmm
      do 7124 ip=2,ihalf-1
      ie=ie-1
      dy(ip)=dy(ip-1)*expy
 7124 dy(ie)=dy(ip)
 7003 continue
c ******* initialize the nodal arrays *******
      do 1111 nd=1,nodc
      su(nd)=0.0
      sv(nd)=0.0
      u(nd)=1.0
      v(nd)=0.0
      uh(nd)=0.0
      vh(nd)=0.0
      du(nd)=0.0
      dv(nd)=0.0
      p(nd)=0.0
      pp(nd)=0.0
      smass(nd)=0.0
      uflx(nd)=0.0
      vor(nd)=1.0
      vol(nd)=0.0
      gam(1,nd)=1.0
      gam(2,nd)=1.0
      do 9313 i=1,3
      do 9312 j=1,3
      a(i,j,nd)=1.0
 9312 continue
 9313 continue
 1111 continue
c
      call geot(ampl,freq)
c ******* set the wall velocities **************
      utop=re
      vtop=0.
      ubot=0.
      vbot=0.
      ulef=0.
      vlef=0.
      urit=0.
      vrit=0.
      vor(np(1,1))=0.
      vor(np(1,jpm))=0.
      vor(np(ipm,1))=0.
      vor(np(ipm,jpm))=0.
      do 1112 j=2,jpmm
      u(np(1,j))=ubot
      v(np(1,j))=vbot
      u(np(ipm,j))=utop
 1112 v(np(ipm,j))=vtop
      do 1113 i=1,ipm
      u(np(i,1))=0.
```

```
      u(np(i,jpm))=0.
      v(np(i,1))=0.
 1113 v(np(i,jpm))=0.
      if(burg) goto 1115
      do 1114 i=1,ipm
      uflx(np(i,1))=1.0
 1114 uflx(np(i,jpm))=0.0
      blk=gr
 1115 continue
      call asmpsi
      call svor
      call lsolve(3,1.0e0,0,3,im2,3,jm2,5,1,1)
c *******************************************************************
      comp=1.0
      if(read) then
      open(15,file='raw.dat',status='old')
      do 2451 n=1,nodc
 2451 read(15,*) vor(n),pp(n),uflx(n)
      close(15)
      endif
c
      do 9000 iterg=0,maxit
      call bvt(bvrel)
      call cofsrt(1)
      if(burg) goto 1116
      call lsolve(5,relt,1,2,im1,3,jm2,2,1,1)
      call tsorc
 1116 continue
      call lsolve(1,relv,1,3,im2,3,jm2,2,1,1)
c ---------------------- convergence check ----------------------
      if(mod(iterg,1).eq.0) then
        resid=0.0
      do 9316 i=2,ipm-1
      do 9315 j=2,jpm-1
        n=np(i,j)
        res=a(1,1,n)*vor(np(i-1,j-1))+a(1,2,n)*vor(np(i-1,  j))+
     1      a(1,3,n)*vor(np(i-1,j+1))+a(2,1,n)*vor(np(i  ,j-1))+
     2      a(2,2,n)*vor(np(i  ,  j))+a(2,3,n)*vor(np(i  ,j+1))+
     3      a(3,1,n)*vor(np(i+1,j-1))+a(3,2,n)*vor(np(i+1,  j))+
     4      a(3,3,n)*vor(np(i+1,j+1))-rhp(np(i,j))
       resid=resid+res*res
 9315 continue
 9316 continue
      write(11,9210) iterg,resid,vor(np(ihalf,jhalf)),
     1pp(np(ihalf,jhalf)),uflx(np(ihalf,jhalf))
      endif
        if(resid.lt.crit) goto 9001
      call svor
      call lsolve(3,1.0e0,0,3,im2,3,jm2,2,1,1)
      do 2033 i=1,ipm
      do 2033 j=1,jpm
 2033 psi(i,j)=pp(np(i,j))
```

```
c ******* write unformatted values of vor,psi,u,v *******
      if(mod(iterg,50).eq.0) then
      call uvclc4
      open(10,file='chk.dat',status='unknown')
      do 7136 j=1,jpm
      do 7136 i=1,ipm
      n=np(i,j)
      write(10,7140) j,i,x(n),y(n),u(n),v(n),uflx(n),vor(n),pp(n)
 7136 continue
      close(10)
      if(write .eq. .true.) then
      open(14,file='raw.dat',status='unknown')
      do 2452 n=1,nodc
 2452 write(14,*) vor(n),pp(n),uflx(n)
      close(14)
      endif
      endif
c **********************************************************
 9000 continue
 9001 continue
      call uvclc4
      write(11,*) '******** t e s t .  for  ********'
      write(11,2440) ngr,ipm,jpm,bvrel,relv,relt
      if(ngr.eq.3) write(11,2441) expx,expy
      write(11,*) 'burg is: ',burg
      write(11,*) 'unifrm is: ',unifrm
      write(11,*) 'the grashof # is: ',gr
      call flux
      write(11,7141)
      do 7135 j=1,jpm
      do 7135 i=1,ipm
      n=np(i,j)
      write(11,7140) j,i,x(n),y(n),u(n),v(n),uflx(n),vor(n),pp(n)
 7135 continue
 7141 format(//,1x,'j',2x,'i',7x,'x',10x,'y',13x,'u',10x,'v',10x,
     +'t',10x,'o',10x,'z',/,('-'))
 7140 format(i2,1x,i2,'!',1x,2(1x,1pe10.3),'!!',1x,5(1x,1pe10.3))
      close(11)
      stop
c **********************************************************
 2440 format(3x,'ngr',2x,'ipm',2x,'jpm',7x,'bvrel',
     12x,' relv',2x,' relt',/,3x,'---',2x,'---',2x,'---',7x,'-----',
     22x,'-----',2x,'-----',/,4x,i1,3x,i3,2x,i3,7x,f5.2,2x,f5.2,2x,
     3f5.2)
 2441 format(//,10x,'expansion coeffs. in x and y',/,10x,28('-'),/,
     112x,f5.2,3x,f5.2)
 9210 format(1x,i4,4x,4(1pd11.4,2x))
      end
c
      subroutine geot(ampl,freq)
      include'comvort.blk'
      dimension yh(20)
```

```
      nodc=ipm*jpm
      numelc=neltc
      neltc=numelc
c
c*** number the nodes first ***
c
      do 800 i=1,ipm
      do 800 j=1,jpm
 800  np(i,j)=(j-1)*ipm+i
      pi=3.1415926536
      ttheta=(1./(4.*freq))/ampl
      trm=1.0
      do 802 j=1,jpmm/2-1
 802  trm=trm*expx+1
      mz=4*freq
      mz2=mz+1
      do 600 n=3,mz2,2
      ny=n-2
      i=(n-1)/2
      yh(i)=float(ny)/float(mz)
 600  continue

      x(1)=0.0
      y(1)=0.0
      dxa1=.5-ampl
      dxa2=.5+ampl
      do 801 i=1,ipm
      if(i.eq.1)then
      yy=0.0
      x(np(i,1))=yy/ttheta
      dx(1)=.5/trm
      x(np(i,1))=0.0
      go to 520
c     go to 400
      endif
      yy=y(np(i-1,1))+dy(i-1)
      if(yy.gt.0.0.and.yy.le.yh(1)) then
      x(np(i,1))=yy/ttheta
      dx(1)=(.5-x(np(i,1)))/trm
      go to 520
      endif
      if(yy.gt.yh(1).and.yy.le.yh(2)) then
      yy1=yy-yh(1)
      x(np(i,1))=yy1/ttheta
      dx(1)=(dxa1+x(np(i,1)))/trm
      x(np(i,1))=-x(np(i,1))+ampl
      go to 520
      endif
      if(yy.gt.yh(2).and.yy.le.yh(3)) then
      yy2=yy-yh(2)
      x(np(i,1))=yy2/ttheta
      dx(1)=(dxa2-x(np(i,1)))/trm
```

```
      x(np(i,1))=x(np(i,1))-ampl
      go to 520
      endif
      if(yy.gt.yh(3).and.yy.le.yh(4)) then
      yy3=yy-yh(3)
      x(np(i,1))=yy3/ttheta
      dx(1)=(dxa1+x(np(i,1)))/trm
      x(np(i,1))=-x(np(i,1))+ampl
      go to 520
      endif
      if(yy.gt.yh(4).and.yy.le.yh(5)) then
      yy4=yy-yh(4)
      x(np(i,1))=yy4/ttheta
      dx(1)=(dxa2-x(np(i,1)))/trm
      x(np(i,1))=x(np(i,1))-ampl
      go to 520
      endif
      if(yy.gt.yh(5).and.yy.le.yh(6)) then
      yy5=yy-yh(5)
      x(np(i,1))=yy5/ttheta
      dx(1)=(dxa1+x(np(i,1)))/trm
      x(np(i,1))=-x(np(i,1))+ampl
      go to 520
      endif
      if(yy.gt.yh(6).and.yy.le.1.0) then
      yy6=yy-yh(6)
      x(np(i,1))=yy6/ttheta
      dx(1)=(dxa2-x(np(i,1)))/trm
      x(np(i,1))=x(np(i,1))-ampl
      go to 520
      endif
 520  dx(jpmm)=dx(1)
      y(np(i,1))=yy
      do 804 j=2,jpmm/2
      dx(j)=dx(j-1)*expx
 804  dx(jpmm-j+1)=dx(j)
      do 805 j=2,jpm
      y(np(i,j))=yy
 805  x(np(i,j))=x(np(i,j-1))+dx(j-1)
 801  continue
c*** now setup the nodal connection array ***
      nb=0
      do 2121 i=1,ipm
      nb=nb+1
 2121 nbnd(nb)=i
      nb1=nb
      do 2122 j=2,jpm
      nb=nb+1
 2122 nbnd(nb)=np(ipm,j)
      nb2=nb
      do 2123 i=ipmm,1,-1
      nb=nb+1
```

```
 2123 nbnd(nb)=np(i,jpm)
      nb3=nb
      do 2124 j=jpmm,2,-1
      nb=nb+1
 2124 nbnd(nb)=np(1,j)
      nbmax=nb
c
      nel=0
      do 9319 iq=1,ipmm
      do 9320 jq=1,jpmm
      nel=nel+1
      nca(nel,1)=np(iq,jq)
      nca(nel,2)=np(iq,jq+1)
      nca(nel,3)=np(iq+1,jq+1)
      nel=nel+1
      nca(nel,1)=np(iq,jq)
      nca(nel,2)=np(iq+1,jq+1)
      nca(nel,3)=np(iq+1,jq)
 9320 continue
 9319 continue
      nel=2*ipmm
      nca(nel,1)=np(ipm,1)
      nca(nel,2)=np(ipmm,2)
      nca(nel,3)=np(ipm,2)
      nel=nel-1
      nca(nel,1)=np(ipm,1)
      nca(nel,2)=np(ipmm,1)
      nca(nel,3)=np(ipmm,2)
      nel=neltc-2*ipmm+1
      nca(nel,1)=np(2,jpmm)
      nca(nel,2)=np(1,jpmm)
      nca(nel,3)=np(1,jpm)
      nel=nel+1
      nca(nel,1)=np(2,jpmm)
      nca(nel,2)=np(1,jpm)
      nca(nel,3)=np(2,jpm)
c
      do 2000 nel=1,neltc
      do 2100 node=1,3
      xt(node)=x(nca(nel,node))
 2100 yt(node)=y(nca(nel,node))
c
      det=xt(1)*yt(2)+xt(2)*yt(3)+xt(3)*yt(1)-
     &    yt(1)*xt(2)-yt(2)*xt(3)-yt(3)*xt(1)
c
      area3(nel)=abs(det/2.0)/3.0
c
      alpha(nel,1)=(yt(3)-yt(2))/det
      alpha(nel,2)=(yt(1)-yt(3))/det
      alpha(nel,3)=(yt(2)-yt(1))/det
c
      beta(nel,1)= (xt(2)-xt(3))/det
```

```
      beta(nel,2)= (xt(3)-xt(1))/det
      beta(nel,3)= (xt(1)-xt(2))/det
c
 2000 continue
c ******* the nodal volumes are assembled here *******
c
      do 2500 nel=1,neltc
      do 2501 n=1,3
c ******* here dv is used to assemble inverse areas for each node **
      dv(nca(nel,n))=dv(nca(nel,n))+1.0/area3(nel)
 2501 vol(nca(nel,n))=vol(nca(nel,n))+area3(nel)
 2500 continue
c ******* the map for the general phi variable *******
      jm1=jmaxc-1
      im1=imaxc-1
      do 9009 j=2,jm1
      do 9009 i=2,im1
 9009 map(i,j)=np(i-1,j-1)
c
      return
      end
c
      subroutine uvclc4
      include'comvort.blk'
c
      dimension ii(3),jj(3)
c
      do 2500 n=1,nodc
      uh(n)=u(n)
 2500 vh(n)=v(n)
      do 2000 n=1,nodc
      u(n)=0.0
 2000 v(n)=0.0
c
      do 1000 nel=1,neltc
      ps1=pp(nca(nel,1))
      ps2=pp(nca(nel,2))
      ps3=pp(nca(nel,3))
c
      ve=alpha(nel,1)*ps1+alpha(nel,2)*ps2+alpha(nel,3)*ps3
      ue=(-1.)*(beta(nel,1)*ps1+beta(nel,2)*ps2+beta(nel,3)*ps3)
c
      do 300 n=1,3
      u(nca(nel,n))=u(nca(nel,n))+ue/(area3(nel)*dv(nca(nel,n)))
 300  v(nca(nel,n))=v(nca(nel,n))+ve/(area3(nel)*dv(nca(nel,n)))
c
 1000 continue
c
c ******* reimpose the boundary velocities ******************
      do 3000 j=1,jpm
      u(np(1,j))=uh(np(1,j))
      v(np(1,j))=vh(np(1,j))
```

```
      u(np(ipm,j))=uh(np(ipm,j))
      v(np(ipm,j))=vh(np(ipm,j))
 3000 continue
c
      do 3001 i=2,ipmm
      u(np(i,jpm))=uh(np(i,jpm))
      u(np(i,1))=uh(np(i,1))
      v(np(i,1))=vh(np(i,1))
      v(np(i,jpm))=vh(np(i,jpm))
 3001 continue
c
      return
      end
c
      subroutine svor
      include'comvort.blk'
c
      do 2000 n=1,nodc
 2000 su(n)=0.0e0
c
      do 1000 nel=1,neltc
      n1=nca(nel,1)
      n2=nca(nel,2)
      n3=nca(nel,3)
      su(n1)=su(n1)+area3(nel)*(22*vor(n1)+7*vor(n2)+7*vor(n3))/36
      su(n2)=su(n2)+area3(nel)*(7*vor(n1)+22*vor(n2)+7*vor(n3))/36
      su(n3)=su(n3)+area3(nel)*(7*vor(n1)+7*vor(n2)+22*vor(n3))/36
 1000 continue
c
      return
      end
c
      subroutine asmpsi
      include'comvort.blk'
c
      dimension ii(3),jj(3)
      do 2000 n=1,nodc
      do 2000 i=1,3
      do 2000 j=1,3
 2000 apsi(i,j,n)=0.0e0
c
      do 1000 nel=1,neltc
      do 100 n=1,3
      xn(n)=x(nca(nel,n))
 100  yn(n)=y(nca(nel,n))
c
      det1=xn(1)*yn(2)+xn(2)*yn(3)+xn(3)*yn(1)-
     &     yn(1)*xn(2)-yn(2)*xn(3)-yn(3)*xn(1)
c
      y23=yn(2)-yn(3)
      y31=yn(3)-yn(1)
      y12=yn(1)-yn(2)
```

```
c
      x32=xn(3)-xn(2)
      x13=xn(1)-xn(3)
      x21=xn(2)-xn(1)
c
      esm(1,1)=y23*y23+x32*x32
      esm(1,2)=y23*y31+x32*x13
      esm(1,3)=y23*y12+x32*x21
      esm(2,2)=y31*y31+x13*x13
      esm(2,3)=y31*y12+x13*x21
      esm(3,3)=y12*y12+x21*x21
c ******* statement of symmetry *******
      esm(2,1)=esm(1,2)
      esm(3,1)=esm(1,3)
      esm(3,2)=esm(2,3)
c
      do 101 i=1,3
      do 101 j=1,3
 101  esm(i,j)=0.5*esm(i,j)/det1
c
c ******* the assembly process onto the apsi matrix *******
      do 120 nc=1,3
      jj(nc)=(nca(nel,nc)-1)/ipm+1
 120  ii(nc)=nca(nel,nc)-ipm*(jj(nc)-1)
      do 121 i=1,3
      nd=np(ii(i),jj(i))
      do 122 j=1,3
      ig=2-(ii(i)-ii(j))
      jg=2-(jj(i)-jj(j))
      apsi(ig,jg,nd)=apsi(ig,jg,nd)+esm(i,j)
 122  continue
 121  continue
c ****************** end assembly ************************
 1000 continue
      return
      end
c
      subroutine tsorc
      include'comvort.blk'
c
      dimension ii(3),jj(3)
      pi=3.141592654
      theta=0.0
      do 2000 n=1,nodc
 2000 rhp(n)=0.0
c
      do 1000 nel=1,neltc
c
      c1=(alpha(nel,1)*uflx(nca(nel,1))+alpha(nel,2)*
     +uflx(nca(nel,2))+alpha(nel,3)*uflx(nca(nel,3)))*area3(nel)
      c2=(beta(nel,1)*uflx(nca(nel,1))+beta(nel,2)*uflx(nca(nel,2))
     1+beta(nel,3)*uflx(nca(nel,3)))*area3(nel)
```

```
c
      sortc=blk*(sin(theta*pi/180.0)*c2-cos(theta*pi/180.0)*c1)
      do 300 n=1,3
      rhp(nca(nel,n))=rhp(nca(nel,n))+sortc
 300  continue
c
 1000 continue
c
c
      return
      end
c
      subroutine lsolve(nvar,rl,ncoef,ists,inds,jsts,jnds,
     1nsweep,nswpx,nswpy)
      include'comvort.blk'
c
      dimension aa(50),bb(50),cc(50),rhs(50)
      nfl=0
      imx=inds-ists+1
      jmx=jnds-jsts+1
      crel=(1.0-rl)/rl
c ******* set phi to 0.0 ************************
       do 4999 i=1,imaxc
       do 4999 j=1,jmaxc
 4999  phi(i,j)=0.0
      if(nvar.eq.3) goto 500
c ********** copy a matrix onto amt **********
      do 5000 n=1,nodc
      do 5000 i=1,3
      do 5000 j=1,3
 5000 ap(i,j,n)=a(i,j,n)
c
      do 5001 n=1,nodc
      coef=2.*(amax1(0.0e0,-a(2,2,n)))*float(ncoef)
      doef=coef+a(2,2,n)
      ap(2,2,n)=doef/rl
      if(nvar.eq.1) rh(n)=rhp(n)+(doef*crel+coef)*vor(n)
      if(nvar.eq.5) rh(n)=(doef*crel+coef)*uflx(n)+du(n)
 5001 continue
      goto 550
 500  continue
c *********** copy the apsi matrix onto amt **********
      do 5002 n=1,nodc
      do 5002 i=1,3
      do 5002 j=1,3
      rh(n)=su(n)
 5002 ap(i,j,n)=apsi(i,j,n)
c
 550  continue
      do 5004 i=2,im1
      do 5004 j=2,jm1
      if(nvar.eq.1) phi(i,j)=vor(map(i,j))
```

```
      if(nvar.eq.5) phi(i,j)=uflx(map(i,j))
      if(nvar.eq.3) phi(i,j)=pp(map(i,j))
 5004 continue
c
c
c ********* start the sweeping *****************
c
      do 2000 nt=1,nsweep
      do 1000 ns=1,nswpx
      ifl=-1
      j=jsts
      do 100 nct=1,2
      j=j+ifl
      ifl=-1*ifl
      do 10 jc=1,jmx
      j=j+ifl
      i=ists-1
      do 1 ic=1,imx
      i=i+1
      n=map(i,j)
      aa(ic)=ap(1,2,n)
      bb(ic)=ap(2,2,n)
      cc(ic)=ap(3,2,n)
      rhs(ic)=rh(n)-ap(1,1,n)*phi(i-1,j-1)-ap(1,3,n)*phi(i-1,j+1)
     1-ap(2,1,n)*phi(i,j-1)-ap(2,3,n)*phi(i,j+1)
     2-ap(3,1,n)*phi(i+1,j-1)-ap(3,3,n)*phi(i+1,j+1)
 1    continue
      rhs(1)=rhs(1)-ap(1,2,map(ists,j))*phi(ists-1,j)
      rhs(imx)=rhs(imx)-ap(3,2,map(inds,j))*phi(inds+1,j)
      call tri(aa,bb,cc,rhs,1,imx)
      i=ists-1
      do 2 ii=1,imx
      i=i+1
 2    phi(i,j)=aa(ii)
 10   continue
 100  continue
 1000 continue
c
      do 1001 ns=1,nswpy
      ifl=-1
      i=ists
      do 101 nct=1,2
      i=i+ifl
      ifl=-1*ifl
      do 11 ic=1,imx
      i=i+ifl
      j=jsts-1
      do 3 jc=1,jmx
      j=j+1
      n=map(i,j)
      aa(jc)=ap(2,1,n)
      bb(jc)=ap(2,2,n)
```

```
      cc(jc)=ap(2,3,n)
      rhs(jc)=rh(n)-ap(1,1,n)*phi(i-1,j-1)-ap(3,1,n)*phi(i+1,j-1)
     1-ap(1,2,n)*phi(i-1,j)-ap(3,2,n)*phi(i+1,j)
     2-ap(1,3,n)*phi(i-1,j+1)-ap(3,3,n)*phi(i+1,j+1)
 3    continue
      rhs(1)=rhs(1)-ap(2,1,map(i,jsts))*phi(i,jsts-1)
      rhs(jmx)=rhs(jmx)-ap(2,3,map(i,jnds))*phi(i,jnds+1)
      call tri(aa,bb,cc,rhs,1,jmx)
      j=jsts-1
      do 4 jj=1,jmx
      j=j+1
 4    phi(i,j)=aa(jj)
 11   continue
 101  continue
 1001 continue
c
 2000 continue
c
      if(phi(5,5).ge.1.0d10)  then
      write(11,*) 'bombing out in nvar=',nvar
      stop
      endif
c ********* resubstitute phi back into the appropriate variable ***
      do 3000 i=ists,inds
      do 3000 j=jsts,jnds
      if(nvar.eq.1) vor(map(i,j))=phi(i,j)
      if(nvar.eq.5) uflx(map(i,j))=phi(i,j)
      if(nvar.eq.3) pp(map(i,j))=phi(i,j)
 3000 continue
c
c
      return
      end
c
      subroutine tri(a,b,c,d,m,n)
      dimension a(50),b(50),c(50),d(50),e(50),f(50),g(50)
c     gauss elimination
      e(m)=b(m)
      f(m)=d(m)
      m1=m+1
      do 10 i=m1,n
      g(i)=a(i)/e(i-1)
      e(i)=b(i)-g(i)*c(i-1)
 10   f(i)=d(i)-g(i)*f(i-1)
c     back substitution. answer stored in a(i)
      a(n)=f(n)/e(n)
      do 20 j=m1,n
      i=n+m1-1-j
 20   a(i)=(f(i)-c(i)*a(i+1))/e(i)
      return
      end
```

```
c
      subroutine shear
      include'comvort.blk'
      dimension shr(100)
      data shr/100*0./
      do 10 j=1,jpmm
      nb=np(ipm,j)
      nbp=np(ipm,j+1)
      el=sqrt((x(np(ipm,j+1))-x(np(ipm,j)))**2+(y(np(ipm,j+1))
     +-y(np(ipm,j)))**2)/2
      shr(j)=shr(j)+dx(j)*(vor(nbp)+3.*vor(nb))/8.
      shr(j+1)=shr(j+1)+dx(j)*(3.*vor(nbp)+vor(nb))/8.
 10   continue
      sum=0.
      do 11 j=1,jpm
 11   sum=sum+shr(j)
      sum=abs(2*sum/(utop**2))
      write(11,*) 'drag coefficient =',sum
      write(11,*) 'shear array follows'
      do 12 j=1,jpm
 12   write(11,13) j,shr(j)
 13   format(4x,i4,3x,d11.4)
      return
      end
c
      subroutine print(ph)
      include'comvort.blk'
c
      dimension ph(nodc)
      jst=-5
      klip=jpm/6+1
      jrem=mod(jpm,6)
      do 7800 k=1,klip
      jst=jst+6
      jnd=jst+5
      if(k.eq.klip) then
      jnd=jpm
      jst=jnd-jrem+1
      endif
      write(11,7900) (x(np(1,j)),j=jst,jnd)
      write(11,7903)
      do 7801 i=ipm,1,-1
 7801 write(11,7901) y(np(i,1)),(ph(np(i,j)),j=jst,jnd)
 7800 continue
 7900 format(//,6x,'x=---->',6(1pe10.3,1x))
 7901 format('y=',1pe10.3,'¦',6(1pe10.3,1x))
 7903 format(79('-'))
      return
      end
c
      subroutine flux
      include'comvort.blk'
```

```
c
      do 1 i=1,imaxc
      do 1 j=1,jmaxc
 1    phi(i,j)=0.
      do 2 i=2,im1
      do 2 j=2,jm1
      phi(i,j)=uflx(np(i-1,j-1))
 2    continue
      do 4 n=1,nodc
 4    rh(n)=du(n)
      sum=0.0
      do 3 n=1,ipm
      ii=n
      jj=1
      i=ii+1
      j=jj+1
      flx=a(2,2,n)*phi(i,j)+a(1,1,n)*phi(i-1,j-1)+a(1,2,n)*
     1phi(i-1,j)+a(1,3,n)*phi(i-1,j+1)+a(2,1,n)*phi(i,j-1)+a(2,3,n)*
     2phi(i,j+1)+a(3,1,n)*phi(i+1,j-1)+a(3,2,n)*phi(i+1,j)+a(3,3,n)*
     3phi(i+1,j+1)-rh(n)
      sum=sum+flx
      write(11,2000) n,flx
      if(n.gt.1 .and. n.lt.ipm) then
      el1sq=(x(n)-x(n-1))**2+(y(n)-y(n-1))**2
      el2sq=(x(n)-x(n+1))**2+(y(n)-y(n+1))**2
      el=(sqrt(el1sq)+sqrt(el2sq))/2
      else if(n.eq.1)then
      el=sqrt(x(2)**2+y(2)**2)/2
      else if (n.eq.ipm)then
      el=sqrt((x(n)-x(n-1))**2+(y(n)-y(n-1))**2)/2
      endif
      uin(n)=flx/el
      write(11,*) 'loc. nuss. no. ',n,uin(n)
 3    continue
      write(11,*) 'net flux=',sum
 2000 format(3x,i4,2x,d11.4)
      return
      end
      subroutine bvt(bvrel)
      include'comvort.blk'
c
      do 1 i=1,imaxc
      do 1 j=1,jmaxc
 1    phi(i,j)=0.
      do 2 i=2,im1
      do 2 j=2,jm1
      phi(i,j)=pp(np(i-1,j-1))
 2    continue
      do 3 nb=1,nbmax
      n=nbnd(nb)
      jj=(n-1)/ipm+1
      ii=n-ipm*(jj-1)
```

```
      i=ii+1
      j=jj+1
      vns=apsi(2,2,n)*phi(i,j)+apsi(1,1,n)*phi(i-1,j-1)+
     1apsi(1,2,n)*phi(i-1,j)+apsi(1,3,n)*phi(i-1,j+1)+apsi(2,1,n)*
     2phi(i,j-1)+apsi(2,3,n)*phi(i,j+1)+apsi(3,1,n)*phi(i+1,j-1)+
     3apsi(3,2,n)*phi(i+1,j)+apsi(3,3,n)*phi(i+1,j+1)
      vn=vns/vol(n)
      vor(n)=(vn-vor(n))*bvrel+vor(n)
 3    continue
      return
      end
c
      subroutine cofsrt(kde)
      include'comvort.blk'
      dimension ii(3),jj(3),xu(3),yu(3),z(10),xb(10),yb(10),
     #          f(10,3),g(10,3),yfxg(3,3),uf(3),vf(3),
     #          ut(10),vt(10)
      do 2000 n=1,nodc
      do 2000 i=1,3
      do 2000 j=1,3
 2000 a(i,j,n)=0.0d0
      do 1000 nl=1,neltc
      n1=nca(nl,1)
      n2=nca(nl,2)
      n3=nca(nl,3)
      ve=alpha(nl,1)*pp(n1)+alpha(nl,2)*pp(n2)+alpha(nl,3)*pp(n3)
      ue=-1.*(beta(nl,1)*pp(n1)+beta(nl,2)*pp(n2)+beta(nl,3)*pp(n3))
      u1=ue
      u2=ue
      u3=ue
      v1=ve
      v2=ve
      v3=ve
      xo=(x(n1)+x(n2)+x(n3))/3.0
      yo=(y(n1)+y(n2)+y(n3))/3.0
      gamma=(gam(1,n1)+gam(1,n2)+gam(1,n3))/3.0
      uav=(u1+u2+u3)/3.0
      vav=(v1+v2+v3)/3.0
      ubav=sqrt(uav*uav+vav*vav)
      cost=uav/ubav
      sint=vav/ubav
      do 9300 i=1,3
      xb(i)=(x(nca(nl,i))-xo)*cost+
     &      (y(nca(nl,i))-yo)*sint
      yb(i)=(y(nca(nl,i))-yo)*cost-
     &      (x(nca(nl,i))-xo)*sint
 9300 continue
      xb(10)=0.0
      yb(10)=0.0
      xb(4)=(xb(1)+xb(2))/2.
      xb(5)=(xb(2)+xb(3))/2.
      xb(6)=(xb(3)+xb(1))/2.
```

```
      xb(7)=xb(4)/2.
      xb(8)=xb(5)/2.
      xb(9)=xb(6)/2.
      yb(4)=(yb(1)+yb(2))/2.
      yb(5)=(yb(2)+yb(3))/2.
      yb(6)=(yb(3)+yb(1))/2.
      yb(7)=yb(4)/2.
      yb(8)=yb(5)/2.
      yb(9)=yb(6)/2.
      cst=rho*ubav/gamma
      xmx=amax1(xb(1),xb(2),xb(3))
      do 9301 i=1,3
      pe=cst*(xmx-xb(i))
      big=amax1(0.,(1.-.1*pe)**5)
      z(i)=big/(pe+big)
      z(i)=(z(i)-1)/cst
 9301 continue
      pe=cst*xmx
      big=amax1(0.,(1.-.1*pe)**5)
      z(10)=big/(pe+big)
      z(10)=(z(10)-1)/cst
      y12=yb(1)-yb(2)
      y31=yb(3)-yb(1)
      y23=yb(2)-yb(3)
      z32=z(3)-z(2)
      z13=z(1)-z(3)
      z21=z(2)-z(1)
      c1=z(2)*yb(3)-z(3)*yb(2)
      c2=z(3)*yb(1)-z(1)*yb(3)
      c3=z(1)*yb(2)-z(2)*yb(1)
      d=z(1)*y23+z(2)*y31+z(3)*y12
      if(d.le.1.00e-08)then
      d=1.00
      endif
      xili=(y23*xb(1)+y31*xb(2)+y12*xb(3))/d
      ximi=(z32*xb(1)+z13*xb(2)+z21*xb(3))/d
      xini=( c1*xb(1)+ c2*xb(2)+ c3*xb(3))/d
      rav=rho*ubav
      do 9302 i=4,10
      f(i,1)=(rav*(z32*yb(i)+c1)-gamma*y23)/d
      f(i,2)=(rav*(z13*yb(i)+c2)-gamma*y31)/d
      f(i,3)=(rav*(z21*yb(i)+c3)-gamma*y12)/d
      g(i,1)=(-gamma*z32)/d
      g(i,2)=(-gamma*z13)/d
      g(i,3)=(-gamma*z21)/d
 9302 continue
      do 9310 j=1,3
      do 9311 i=1,3
      yfxg(j,i)=((f(j+3,i)+4.*f(j+6,i)+f(10,i))*yb(j+3)
     #          -(g(j+3,i)+4.*g(j+6,i)+g(10,i))*xb(j+3))/6.
 9311 continue
 9310 continue
```

```
      do 9320 i=1,3
      esm(1,i)=yfxg(3,i)-yfxg(1,i)
      esm(2,i)=yfxg(1,i)-yfxg(2,i)
      esm(3,i)=yfxg(2,i)-yfxg(3,i)
9320  continue
      do 120 nc=1,3
      jj(nc)=(nca(nl,nc)-1)/ipm+1
120   ii(nc)=nca(nl,nc)-ipm*(jj(nc)-1)
      do 121 i=1,3
      nd=np(ii(i),jj(i))
      do 122 j=1,3
      ig=2-(ii(i)-ii(j))
      jg=2-(jj(i)-jj(j))
      a(ig,jg,nd)=a(ig,jg,nd)+esm(i,j)
122   continue
121   continue
1000  continue
      return
      end
```