Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

The Van Houten library has removed some of the personal information and all signatures from the approval page and biographical sketches of theses and dissertations in order to protect the identity of NJIT graduates and faculty.

INFORMATION TO USERS

This reproduction was made from a copy of a document sent to us for microfilming. While the most advanced technology has been used to photograph and reproduce this document, the quality of the reproduction is heavily dependent upon the quality of the material submitted.

The following explanation of techniques is provided to help clarify markings or notations which may appear on this reproduction.

1. The sign or "target" for pages apparently lacking from the document photographed is "Missing Page(s)". If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting through an image and duplicating adjacent pages to assure complete continuity.

2. When an image on the film is obliterated with a round black mark, it is an indication of either blurred copy because of movement during exposure, duplicate copy, or copyrighted materials that should not have been filmed. For blurred pages, a good image of the page can be found in the adjacent frame. If copyrighted materials were deleted, a target note will appear listing the pages in the adjacent frame.

3. When a map, drawing or chart, etc., is part of the material being photographed, a definite method of "sectioning" the material has been followed. It is customary to begin filming at the upper left hand corner of a large sheet and to continue from left to right in equal sections with small overlaps. If necessary, sectioning is continued again—beginning below the first row and continuing on until complete.

4. For illustrations that cannot be satisfactorily reproduced by xerographic means, photographic prints can be purchased at additional cost and inserted into your xerographic copy. These prints are available upon request from the Dissertations Customer Services Department.

5. Some pages in any document may have indistinct print. In all cases the best available copy has been filmed.

University Microfilms International
300 N. Zeeb Road
Ann Arbor, MI 48106

8317604

Wu, Chicheng

MOTION OF VISCOUS LIQUID IN ROTATING CYLINDER

New Jersey Institute of Technology D.ENG.SC. 1983

University Microfilms International 300 N. Zeeb Road, Ann Arbor, MI 48106

MOTION OF VISCOUS LIQUID

IN ROTATING CYLINDER

by
ChiCheng Wu

Dissertation submitted to the Faculty of the Graduate School
of the New Jersey Institute of Technology in partial fulfillment
of the requirements for the degree of
Doctor of Engineering Science
1982

APPROVAL SHEET

Title of the Thesis: Motion of Viscous Liquid in Rotating Cylinder

Name of Candidate: ChiCheng Wu
Doctor of Engineering Science, 1982

Thesis and Abstract Approved: ______________________ ________
Richard C. Progelhof Date
Professor
Department of Mechanical Engineering

______________________ ________
Date

Signatures of other members ______________________ ________
of the thesis committee. Date

______________________ ________
Date

______________________ ________
Date

VITA

Name: ChiCheng Wu

Degree and Date to be Conferred: D. Eng. Sc., 1983

Secondary Education: Changhua High School, 1959

Collegiate Institutions Attended	Dates	Degree	Date of Degree
Tunghai University	9/59-6/63	B.S.ch.E.	6/63
University of Missouri-Rolla	9/69-8/70	M.S.ch.E.	8/70
New Jersey Institute of Technology	9/72-6/82		

Major: Chemical Engineering

Publications:

"Gamma-Ray Induced Polymerization of Unsaturated polyester in Red Mud" Presented at Symposium on the Utilization of Large Radiation Sources and Accelerators in Industrial Processing - International Atomic Energy Agency, Munich, Germany, August, 1969.

Positions Held:

Process Design Engineer, American Cyanamid Co., Pearl River, N.Y.

System Engineering, M. W. Kellogg Co., Hackensack, N.J.

Process Engineering, Lummus Co., Bloomfield, N.J.

Chemical Operator, Pitt-Consol Chemical Co., Newark, N.J.

Chemical Engineer, Marks Polarized Corp., New York, N.Y.

Chemical Engineer, Union Industrial Research Institute, Taiwan, China.

Chemical Engineer, Taiwan Alkali Co., Taiwan, China.

ABSTRACT

A computer model for the withdrawl and buildup of the fluid from the pool bounded on the horizontal, rotating, cylindrical surface has been developed. For a imcompressible Newtonian fluid, the model is based upon the modified SOLA-SURF algorithm to solve the finite-difference form of Navier-Stokes equations on primitive variables. A hybird of centered and partial donor cell differencings is applied to the momentum convective flux. In each time cycle, the cell velocity is adjusted by the pressure corrector and iterates to satisfy the equation of continuity. The free surface position is approximated by a surface height function derived from kinematic equation. The flow field moves under control of constant fluid volume. When the bottom boundary is a free surface, it is necessary to make good initial guess of the velocities for the newly entered cells.

An example computation predicts the onset velocity of rimming flow within 5% of that of an empirical correlation for unsaturated polyester. It indicates the correlation can be applied to the fluid of medium range viscosity. The fluid velocities in the vicinity of rigid boundary are close to rotating velocity. Backward velocities occur near the free surface. The transient flow phenomena is qualitatively in agreement with the experimental observations. The criterion of constant fluid volume is held within 10%. Singularity resulted in the calculation may be an indication that the assumption of no-slip boundary condition at contact line is invalid. The size of time step is found very critical to the stability of computation.

TABLE OF CONTENTS

TABLE OF CONTENTS Cont'd

LIST OF FIGURES

LIST OF FIGURES Cont'd

LIST OF FIGURES Cont'd

LIST OF TABLES

I. INTRODUCTION

This study was undertaken to develop a computer model to predict the incompressible, viscous, Newtonian fluid in a horizontal cylinder rotating about the major axis, a simulation of reactive rotational molding of pipe.

I.A Rotational Reactive Molding

Two types of rotational molding processes are encountered commercially, thermoplastic powder and reactive liquid. The largest quantity of product is made with the thermoplastic system. In the powder thermoplastic system, the hollow split mold is, at the beginning, charged with the ground thermoplastics, e.g., polyethylene. Then, the mold is locked close and the powder is heated until it becomes a homogeneous melt, while the mold is rotating. The size of ground powder is usually smaller than 35 mesh, but varied according to material flow properties and product shape.

Since the process involves a transient solid-liquid boundary not definable by simple function of coordinates and heat transfer is equally as important as momentum transfer, it is beyond the scope of this study. The reactive system is similar with the exception of a viscous liquid being used as the charge. It is this system that this work is directed. The horizontal cylinder is the geometry encountered when rotational molding of reactive liquid system for manufacturing pipe. Thus, the term rotational molding will be used in this work to refer to the reactive liquid system in the following discussion.

I.A.1 Technology

The basic operation of reactive rotational molding includes three steps. The schematic diagram of the steps is shown in Fig. 1.

1. Loading

Materials of up to five components (1, 2) are placed in a two-piece mold.

a. Resin - Thermosets, e.g., polyester, epoxy, and urethane
b. Curing agent - copolymers containing double bonds, e.g., styrene and diallyl phthalate for polyester, and amines for epoxy
c. Catalyst (initiator) - Peroxides, e.g., methyl ketone peroxide and benzoyl peroxide
d. Promoter (accelerator) - Compounds to reduce the curing time at room temperature, e.g., cobalt naphthenate, cobalt octoate and p-toluenesulfonic acid
e. Filler - Any material, e.g., glass fiber, cotton, wood flour, clay, coloring matter, etc., to enhance physical properties or to reduce the cost

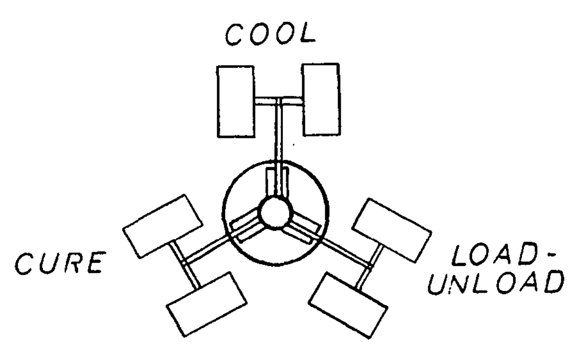


Fig. 1 Typical three-arm rotational molding unit shown in cross-section.

2. Curing

In general, the mold is rotated biaxially about its equatorial and polar axes. For cylindrical geometries the rotation is limited to a single axis, the center of the cylinder. The liquid withdraws from the pool and coats the inner surface of the mold with curing undergoing simultaneously. Curing is a free-radical cosslinking polymerization, causing resin to change from a liquid to solid. This process results in an increase in the viscosity of the resin.

During the process, two stages of phenomenal change can be recongnized (2).

a. Setting time (Gel time) - From first initiation of free radicals to formation of soft gel

b. Maturing time - From a soft gel to the final fully cured state

The rotating speed should be set such that the liquid resin uniformly coats the entire mold surface within the gel or setting time.

Curing is an exothermic reaction. The amount of heat liberated depends upon the type of resin, degree of unsaturation and type of copolymer. The heat release rate varies according to temperature, catalyst and promoter. After a physical dimension and a chemical system are selected, temperature becomes the only process variable. As noted in other polymerization processes, curing causes the resin to densify and the part will shrink. However, the heat of polymerization raises the temperature and stretches the polymer chain. If the temperature goes higher and higher, the polymer chain will extend more and more. On the other hand, the curing rate will increase further and cause additional shrinkage. Such a condition may cause cracks. A high temperature also increases mobility of the fluid. A spotty weakened molding may be resulted. Because the mold is rotating, bulky, and enclosed, it is a difficult task to control the curing temperature by its surrounding environment. Frequently, it is the choice of chemical system that solves the control problem. The following factors are the most important considerations (1, 2):

a. Degree of saturation - The more unsaturation present in the polymer, the higher the heat of curing, and the greater the shrinkage. A resin with high degree of unsaturation should cure at low temperature. The catalyst should accelerate the reaction at only moderate rate. When a high temperature and high reaction rate become necessary for a highly unsaturated resin, greater ratio of filler to resin may be added to absorb the heat and reduce the shrinkage so that cracking can be prevented.

b. Size of batch - Resin is a poor conductor of heat. The temperature rise will increase as the batch size increases. Other than selecting a low temperature and a moderate catalyst, it is helpful to adjust the rotating speed so that the liquid coats the whole cylinder in a short time. This will increase the heat transfer area and help to uniformly spread the exotherm.

c. Conductivity of mold - A metal mold has much greater thermal conductivity than a mold made from plastics. Aluminum and copper are most widely used materials for mold construction. When a plastic mold is used for development work, the mold is usually metalized or the plastic impregnated with metalic fillers to increase conductivity, and remove heat from the system to the surroundings.

3. Unloading

The final part can either be removed from the mold in a hot condition or after some cooling. The cooling step lengthens the cycle time. However, it provides better dimensional stability and improves properties.

I.A.2 Advantages

Rotational molding is primarily intended for the manufacture of larger objects usually made of metal and unable to be made by other plastic molding operations, e.g., injection molding, blow molding, etc. The advantages of the rotational molding process include (3):

1. Low equipment cost - It does not involve high operating pressure, high shear rate or precise metering of materials. The mold and tools are relatively low-cost and long-lasting.
2. Low operating cost - It does not involve high energy consumption for heating and compression. The scrap loss is minimal.
3. High production flexibility - It is conducive to low volume production of large articles. But it can be used to rotomold a large number of small items at a time, e.g., vinyl squeeze ear syringes. Simultaneous production of various products in one mold

is possible. It is easy to change materials and colors.

4. Special product quality - The resulting product has low residual molding stress, because no pressure is applied to the resin during process.

This study will focus on the fluid mechanics of withdrawal of the reactive liquid from the pool before curing starts. The analysis will also be limited to uni-axial rotation. The assumption is that extrapolation from one dimensional to three dimensional model is applicable in the liquid rotational molding as in powder thermoplastics molding (4).

I.B. Flow Phenomena

The flow phenomena of a horizontal rotating cylinder (5) are shown in Fig. 2. The process begins with a stationary cylinder with a pool of fluid on the bottom of the cylindrical cavity as shown in Fig. 2A. As the cylinder is rotated, the fluid adheres to the metal cylinder. Various flow regimes will be developed depending upon the angular velocity and fluid properties.

1. Withdrawl - At low rpm, a portion of liquid is drawn onto the wall, Fig. 2B.
2. Cascading - As rpm increases, the pool keeps moving upward. Some of the liquid releases from the wall and falls back to the bottom, Fig. 2C.
3. Rimming - Further increasing angular rotational velocity, the liquid is carried completely around the cylinder, Fig. 2D.
4. Collapsing - If the angular rotational velocity is decreased significantly from rimming velocity to an unstable flow, the fluid is again released to fall to the bottom, Fig. 2E.

5. Hydrocyst (Disking) - At angular velocity above that of rimming, stable thin disks perpendicular to rotating axis are formed. The disks may shift and appear at regualr intervals, Fig. 2G. There is no evidence up to now that disking occurs in biaxial rotation (6), nevertheless this is the secondary flow phenomenon to be studied and avoided in the process.
6. Solid Body Rotation (SBR) - The angular rotational velocity is further increased and makes the liquid uniformly coat the cylinder, Fig. 2F.
7. Pool Rotation - If the angular rotational velocity is increased rapidly from a stationary condition, the pool will rotate, Fig. 2F.

There are other secondary fluid flow phenomena that also occur, e.g., air pumping and air entrainment. Air pumping is a phenomenon where an air layer is developed between the pool and the cylinder surface. Air entrainment is a phenomenon where air bubbles are encapsulated in the fluid layer by physical mixing. These bubbles are well distributed in the liquid.

It should be noted that the primary flow phenomena -- withdrawl, cascading, rimming and SBR -- can be found in any fluid. The secondary phenomena may be observed only in some fluids with a certain range of viscosity. For instance, the collapsing phenomenon happens mostly with low viscosity liquids such as water. Air pumping is, however, observed in highly viscous liquids.

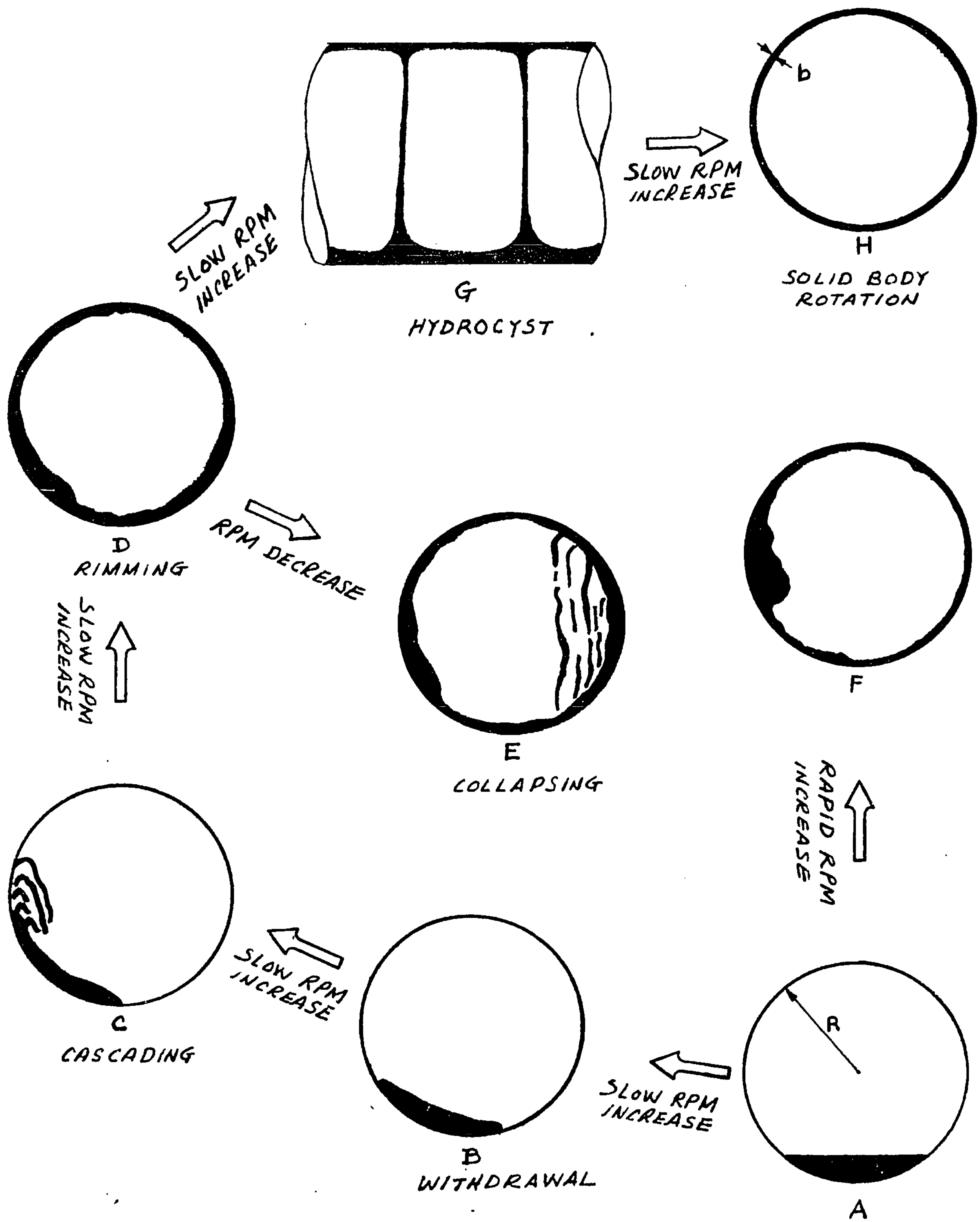


Fig. 2 Various flow phenomena for pool starting to rotate from quiescent state A

II THEORY

The general forms of continuity and momentum equations are the basis of viscous flow in a partially liquid-filled rotating cylinder. To analyze these equations coupled with initial and rigid boundary conditions has some difficulties (25):

1. The location of the free surface is supposed to be a boundary condition, but it is changing and not known a priori.
2. The shape of the free surface has nonlinear effects on the flow.
3. A stress singularity may be resulted at the intersection of free surface and rigid boundary.

Therefore, some investigators have directed their efforts to experimental examination, which is believed more likely to gain a full understanding of the transient process. However, theoretical work is necessary for interpreting the experimental results. The following is a summary of both the experimental and theoretical studies.

II.A. Analysis Of Previous Experimental Data

Progelhof et al. (5-11) analyzed previous experimental results (13- 18) and suggested a simple correlation for each flow regime.

For polyester with viscosity in the range of 200 - 7400 centipoise

Cascading: $$Fr_m = 0.8\ (Re_m)^{0.867} \quad (1)$$

Rimming: $$Fr_m = 1.11\ (Re_m)^{0.867} \quad (2)$$

Solid Body Rotation: $$Fr_m = (2.37 - 0.026\nu)\ (Re_m)^{0.5} \quad (3)$$

Hydrocyst: $$Fr_m = 0.285\ (R/b)\ (Re_m) \quad (4)$$

For water-like liquids with viscosity less than 2 centipoise

Cascading: $$Fr = 243\ (Re)^{0.167}\ (1 - b/R)^{41.9} \quad (5)$$

Rimming: $$Fr = 160\ (Re)^{0.19}\ (1 - b/R)^{21.5} \quad (6)$$

Collapsing: $$Fr = 249\ (Re)^{0.062}\ (1 - b/R)^{31.4} \quad (7)$$

where Fr = Froude Number, $\omega^2 R^2/gb$

Fr_m = Modified Froude Number, $\omega^2 R/g$

Re = Reynolds Number, $\omega Rb/\nu$

Re_m = Modified Reynolds Number, $\omega b^2/\nu$

b = Average film thickness, cm

g = Gravitational acceleration, cm/sec^2

R = Diameter of cylinder, cm

ω = Speed of rotation, radians/sec

ν = Kinematic viscosity, cm^2/sec

Recently, the interest in secondary flow behaviors has generated some data leading to conflicting correlations to describe the hydrocyst regime. When free surface effect was excluded, Progelhof et al. (7) proposed the following equations:

$$J = K_1 \qquad Re_m < K_3 \tag{8a}$$

$$J = K_2 Re_m^N \qquad Re_m > K_3 \tag{8b}$$

where J = Jay Number, $gb/\omega\nu$

K_1, K_2, K_3, N = constants

For polyester, $K_1 = 3.51$, $K_2 = 2.35$, $K_3 = 1.5$, $N=1$

The dimensionless group, Jay number, is defined as ratio of gravitational force ($b^3 \rho g/g_c$) to viscous force ($b^2 \eta w/g_c$). Balmer and Wang (18) concluded a complex correlation which was rearranged into the following form (8) in contrast with Eqs. 8a and 8b.

$$J = 2.995\,(10^{-6})\,Ca^3\,(1-Vf)^{-1.216}\,(\eta/\rho^2)_{air}\,/\,(\eta/\rho^2) \tag{9}$$

where Ca = Capillary Number, $R\omega\eta/\sigma$

Vf = Volume fraction of liquid in cylinder

η = Absolute Viscosity, dyne sec/cm^2

ρ = Density, gm/cm^3

σ = Surface tension, dynes/cm

II.B. Solution By Analytical Methods

Phillips (19) analyzed the momentum and continuity equations for an inviscid fluid by small perturbation theory. A criterion for stability of flow in a horizontal cylinder is found under steady gravity-induced disturbance:

$$\omega^2 R/g > 3/(1 - b/R) \qquad (10)$$

The perturbation approximation is inapplicable under the following three conditions: (1) the film thickness is very small, (2) the flow is viscosity controlled, and (3) a wave motion of large amplitude is present. This criterion is equivalent to collapsing velocity and in agreement with other experimental data (13 - 15) based on water. Phillip's stability condition is shown in Fig. 3.

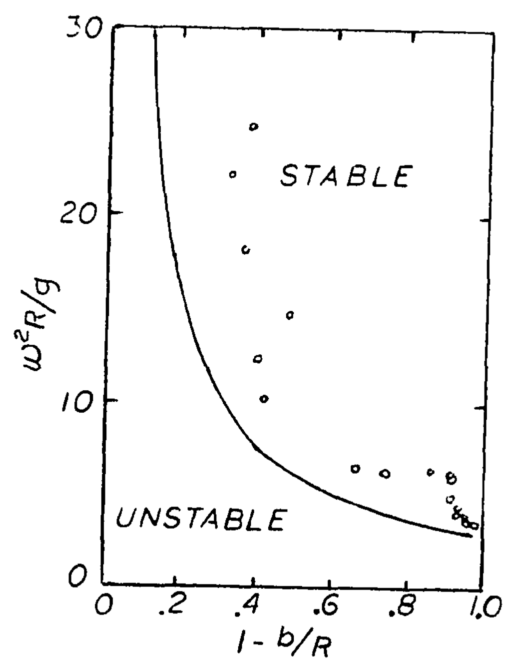


Fig. 3. Phillips' stability criterion.

(experimental data: water)

Cerro and Scriven (20) numerically solved the continuity and momentum equations for solid body rotation. With the condition of $b/R \ll 1$ (typical in rotational molding), a boundary layer approximation can be applied to these equations. Because of the unknown free surface position, the computation in a real coordinate system was very sensitive to minor changes in the boundary conditions and required considerable iterations to obtain the steady state solution.

Ruschak and Scriven (21) analyzed the steady flow as a regular perturbation from rigid body rotation. The continuity and momentum equations are transformed in terms of stream function and four dimensionless groups including Reynolds number, Froude number, Weber number and liquid loading. The general solutions is in a form a Bessel functions. Limiting cases are studied for extreme Reynolds numbers (Re_m) and small b/R. The location of thickest film shifts in the upper quadrant on the rising side, as shown in Fig. 4. The location of thinnest film is diametrically opposite.

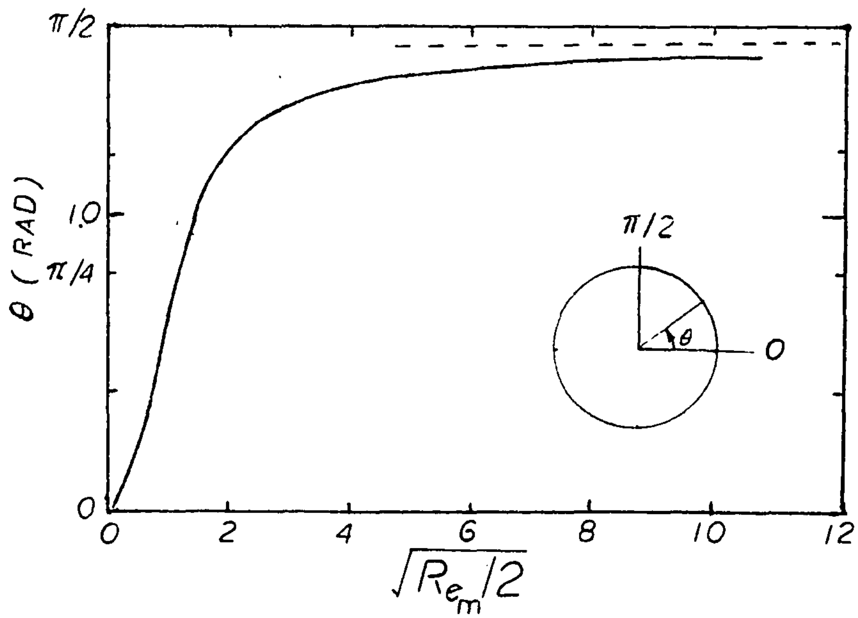


Fig. 4 Angular position, θ (in radians) of thickest film as function of Reynolds number, Re_m, in the limit $b/R \to 0$ when Fr_m is large.

Greenspan (22) also used perturbation methods to study a variant of this problem in which the free surface is replaced by a weightless mobile rigid surface. In terms of the Jacobian and stream functions, the equation is primarily solved for a rapidly rotating flow with large b/R ratio. The onset of instability is compared with Phillips' criterion in Fig. 5.

Gans (23) studied the effects of viscosity, nonlinear interaction and finite cylinder length on flow stability for a large Froude number (Fr_m) and small Ekman number (E) which is defined as $\nu/\omega R^2$. His results were comparable with Greenspan's observation (22) and Philipps' collapsing criterion (19).

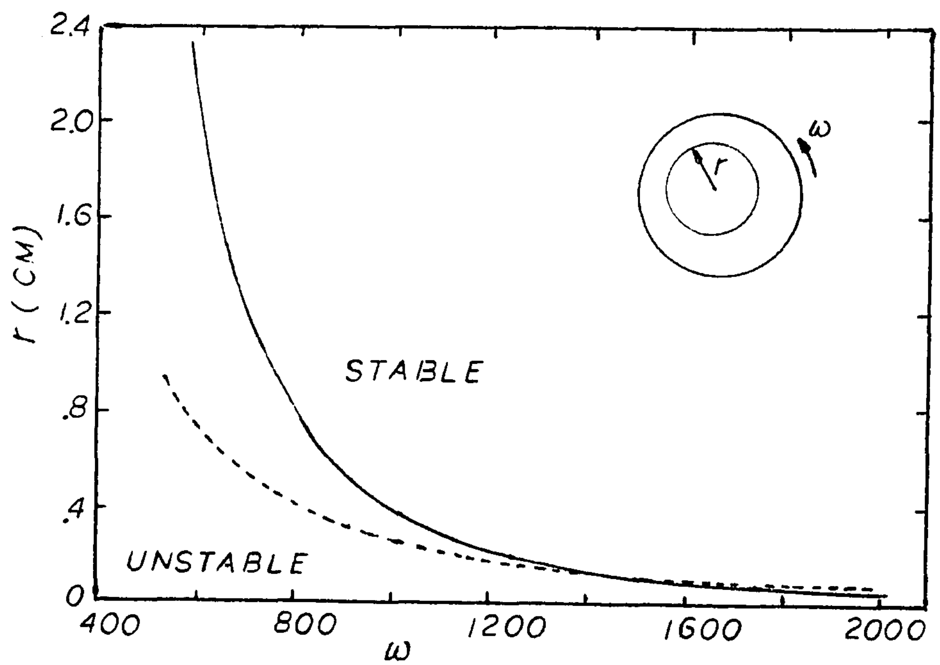


Fig. 5 Greenspan's stability criterion (solid curve) comparing Phillips' criterion (dashed curve).

Deiber and Cerro (24) transformed the continuity and momentum equations to a streamline coordinate system through a modified von Mises transformation. Using finite difference scheme, the numerical method is able to obtain a steady-state solution. It can predict both film thickness and velocity profile of rimming flow. It proposed also the following correlation of solid body rotation (7):

$$Fr_m = Re_m \tag{11}$$

which is different from Eq. 2. When the laminar circular streamline flow is broken, e.g., the fluid has a negative azimuthal velocity, the transformation is invalid. This region is defined discontinuous solution region which coincides the collapsing velocity. The general results are shown in Fig. 6.

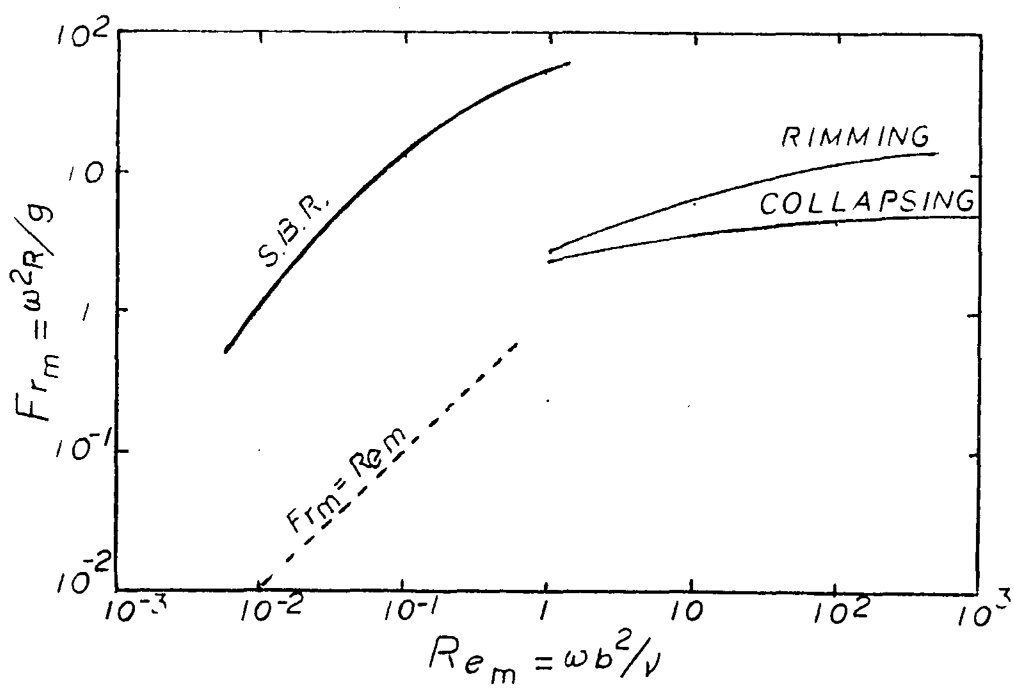


Fig. 6 Flow regimes of Deiber and Cerro, as funciton of Reynolds and Froude numbers.

Orr and Scriven (25) applied a finite element method to a primitive form, i.e., variables of velocity and pressure, instead of the vorticity-stream function system, of the continuity and momentum equations. A numerical solution of steady rimming flow which includes surface tension is developed. The elements need not have uniform size and shape. The Hermite cubic triangular element of Zlamal is chosen. The trial functions are cubic polynomials including ten terms of r and θ. To determine the location of free surface, the residual in the normal-stress boundary condition is used as the criterion. One example of the computation is shown in Fig. 7.

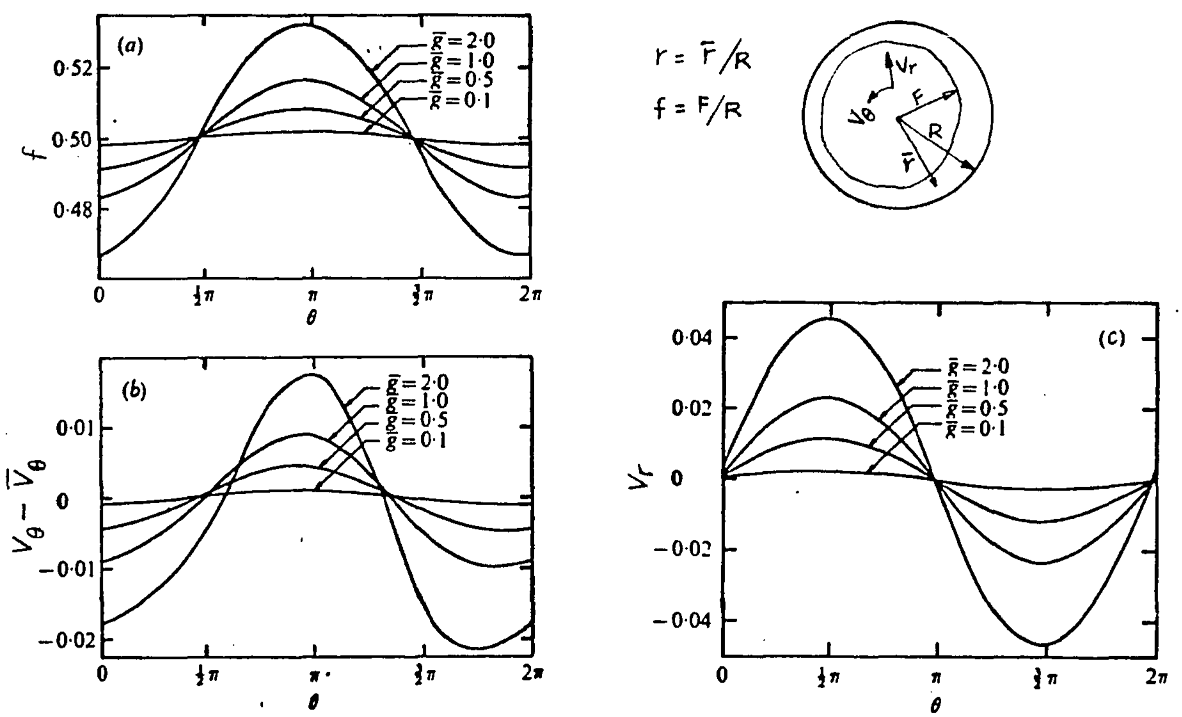


Fig. 7 Results of finite-element method, $Re_m\ (R/b\ )^2 = 1$. Illustrating the effect of increasing gravity on (a) the free-surface location, (b) the departure of the free-surface azimuthal velocity from SBR, $V_\theta - \bar{V}_\theta$, and (c) the radial velocity at free surface, Vr, at various angular positions, θ, with $\bar{g} = g/w^2R = 1/Fr_m$.

The main drawback of the finite element method is that there is no clear indication of optimal combined choice of elements and trial functions.

All of the above mentioned analytical methods start with the continuity and momentum equations. Perturbation methods have proved useful for some cases of steady solutions. But numerical methods are capable to handle more varieties of the problem. The efforts also have been directed to steady state solutions. In the following analysis the transient solution will be developed.

III FORMULATION

A transient solution of a viscous liquid flow in a horizontal rotating cylinder is quite different from a steady rimming flow. Neither of the two features for steady flow mentioned by Orr and Scriven (25) is applicable. Those two features are (1) neither an inflow nor an outflow boundary existing and (2) no gas/liquid/solid contact line. The first feature avoids the complicated boundary conditions. The second one avoids the possibility of a singularity. The flow in this work is idealized by assuming the following conditions:

1. The fluid is incompressible and Newtonian.
2. The effect of surface tension is excluded.
3. The cylinder length is infinite. (No end effects are considered).

III.A Differential Equations

For a Newtonian, incompressible fluid at constant temperature, the momentum equations are the Navier-Stokes equations, which are usually expressed in vector form:

$$\frac{D\vec{u}}{Dt} = -\frac{1}{\rho}\nabla\bar{P} + \frac{\eta}{\rho}\nabla^2\vec{u} + \vec{g} \tag{12a}$$

where

ρ = Density, g/cm^3

$\bar{p}$ = Pressure, dyne/cm^2

η = Newtoniam viscosity, g/cm sec

The equation of continuity is

$$(\nabla\cdot\vec{u}) = 0 \tag{13a}$$

By combining with Eq. 13a, Eq. 12a can be transformed into

$$\frac{\partial\vec{u}}{\partial t} + [\nabla\cdot\vec{u}\vec{u}] = -\frac{1}{\rho}\nabla\bar{P} + \frac{\eta}{\rho}\nabla^2\vec{u} + \vec{g} \tag{12b}$$

Eq. 12b is the equation of motion that can be derived directly from momentum balance. Therefore, its finite difference form retains explicit momentum conservation. However, the difference form of Eq. 12a does not (34). The principle of momentum conservation allows no internal contributions to the time rate of momentum change in a space interval. The contrast in these two equations can be realized by summing over many cells in the difference forms of the one-dimensional equations of Eqs. 12a, and b, for example,

$$(u_i^o - u_i)/\delta t + [\, u_i(u_{i+\frac{1}{2}} - u_{i-\frac{1}{2}}) + (P_{i+\frac{1}{2}} - P_{i-\frac{1}{2}})/\rho \,]/\delta x = 0 \qquad (12a')$$

$$(u_i^o - u_i)/\delta t + [\, (u^2_{i+\frac{1}{2}} - u^2_{i-\frac{1}{2}}) + (P_{i+\frac{1}{2}} - P_{i-\frac{1}{2}})/\rho \,]/\delta x = 0 \qquad (12b')$$

In the summation form, all the flux terms in Eq. 12b' can be cancelled in pairs, while some terms in Eq. 12a' can not. Since the finite difference approximation can not calculate space derivatives precisely, it becomes necessary to secure a better accuracy of approximation by using Eq. 12b instead of Eq. 12a as the starting point for deriving the difference equations. In Cartesian coordinates Eq. 12b can be written as

$$\frac{\partial u}{\partial t} + \frac{\partial(uu)}{\partial x} + \frac{\partial(uv)}{\partial y} + \frac{\partial(uw)}{\partial z} = -\frac{\partial p}{\partial x} + \nu \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right] + g_x \qquad (12c)$$

$$\frac{\partial v}{\partial t} + \frac{\partial(vu)}{\partial x} + \frac{\partial(vv)}{\partial y} + \frac{\partial(vw)}{\partial z} = -\frac{\partial p}{\partial y} + \nu \left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} \right] + g_y \qquad (12d)$$

$$\frac{\partial w}{\partial t} + \frac{\partial(wu)}{\partial x} + \frac{\partial(wv)}{\partial y} + \frac{\partial(ww)}{\partial z} = -\frac{\partial p}{\partial z} + \nu \left[\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \right] + g_z \qquad (12e)$$

The continuity equation is also expressed in Cartesian coordinates, as

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \qquad (13)$$

where u = X-direction velocity, cm/sec

v = Y-direction velocity, cm/sec

w = Z-direction velocity, cm/sec

p = Ratio of pressure to density, $\bar{p}/\rho$, cm^2/sec^2

ν = Kinematic viscosity, η/ρ , cm^2/sec

g_x, g_y, g_z = Gravitational acceleration in X, Y, Z direction, respectively, cm/sec^2

Consider an infinitely long cylinder of radius R rotating around its Z axis (Fig. 8). Under this assumption the fluid will have no velocity component or functionality in the Z direction. Thus, 3-dimensional equations are simplified to a 2-dimensional system by eliminating all the z terms.

$$\frac{\partial u}{\partial t} + \frac{\partial (uu)}{\partial x} + \frac{\partial (uv)}{\partial y} = -\frac{\partial p}{\partial x} + \nu \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right] + g_x \qquad (14a)$$

$$\frac{\partial v}{\partial t} + \frac{\partial (vu)}{\partial x} + \frac{\partial (vv)}{\partial y} = -\frac{\partial p}{\partial y} + \nu \left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right] + g_y \qquad (14b)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \qquad (15)$$

These equations with initial and boundary conditions have not as yet been solved analytically. The numerical approximation methods, e.g., finite difference, can provide a practical solution.

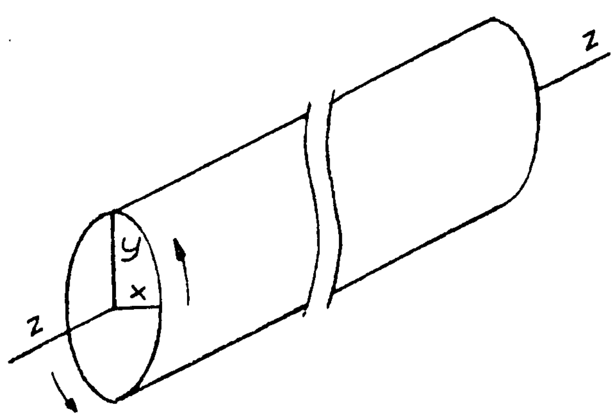


Fig. 8 Cartesian coordinates for a rotational cylinder.

III.B Finite Difference Method

The finite difference method is based upon elementary representations of the derivatives of a smooth function in terms of appropriate difference quotients (26). The Taylor series expansion is the basis of the approximation. For example, a function of two independent variables is expanded as:

$$u(x+\Delta x, y+\Delta y) = u(x,y) + \frac{\partial u(x,y)}{\partial x}\Delta x + \frac{\partial u(x,y)}{\partial y}\Delta y$$
$$+ \frac{\partial^2 u(x,y)}{2\,\partial x^2}\Delta x^2 + \frac{\partial^2 u(x,y)}{\partial x\,\partial y}\Delta x\Delta y + \frac{\partial^2 u(x,y)}{2\,\partial y^2}\Delta y^2$$
$$+ 0(|\Delta x|^3 + |\Delta y|^3) \qquad (16)$$

where $0(|\Delta x|^3 + |\Delta y|^3)$ = Reminder of third order of magnitude

The first partial derivatives $\partial u/\partial x$ and $\partial u/\partial y$ are obtained by setting $\Delta y = 0$ and $\Delta x = 0$, respectively.

$$\frac{\partial u(x,y)}{\partial x} = \frac{u(x+\Delta x, y) - u(x,y)}{\Delta x} + 0(\Delta x) \qquad (17a)$$

$$\frac{\partial u(x,y)}{\partial y} = \frac{u(x,y+\Delta y) - u(x,y)}{\Delta y} + 0(\Delta y) \qquad (17b)$$

The difference approximation shown above is between the forward point and the original point. These representations of the derivative are referred to as forward differencing. The approximation can also be derived from backward differencing in which the Taylor expansion is given by

$$u(x-\Delta x, y-\Delta y) = u(x,y) - \frac{\partial u(x,y)}{\partial x}\Delta x - \frac{\partial u(x,y)}{\partial y}\Delta y$$
$$+ \frac{\partial^2 u(x,y)}{2\,\partial x^2}\Delta x^2 + \frac{\partial^2 u(x,y)}{\partial x\partial y}\Delta x\Delta y$$
$$+ \frac{\partial^2 u(x,y)}{2\,\partial y^2}\Delta y^2 + 0(|\Delta x|^3 + |\Delta y|^3) \qquad (18)$$

Then, the first derivatives are obtained by setting $\Delta y = 0$ or $\Delta x = 0$.

$$\frac{\partial u(x,y)}{\partial x} = \frac{u(x,y) - u(x-\Delta x,y)}{\Delta x} + 0\ (\Delta x) \tag{19a}$$

$$\frac{\partial u(x,y)}{\partial y} = \frac{u(x,y) - u(x,y-\Delta y)}{\Delta y} + 0\ (\Delta y) \tag{19b}$$

If Eq. 16 and Eq. 18 are added and $\Delta y = 0$ or $\Delta x = 0$ is set, the second derivatives are found

$$\frac{\partial^2 u(x,y)}{\partial x^2} = \frac{u(x+\Delta x,y) - 2u(x,y) + u(x-\Delta x,y)}{\Delta x^2} + 0\ (\Delta x^2) \tag{20a}$$

$$\frac{\partial^2 u(x,y)}{\partial y^2} = \frac{u(x,y+\Delta y) - 2u(x,y) + u(x,y-\Delta y)}{\Delta y^2} + 0\ (\Delta y^2) \tag{20b}$$

However, if combining Eqs. 16 and 18 without setting Δx or Δy equal to zero, the expression of mixed derivative $\partial^2 u/\partial x \partial y$ will yield

$$\frac{\partial^2 u(x,y)}{\partial x \partial y} \Delta x \Delta y = \tfrac{1}{2} \Big[u(x+\Delta x,y+\Delta y) + u(x-\Delta x,y-\Delta y) - 2u(x,y)$$

$$- \frac{\partial^2 u(x,y)}{\partial x^2} \Delta x^2 - \frac{\partial^2 u(x,y)}{\partial y^2} \Delta y^2 \Big] + 0\ (\Delta x^4 + \Delta y^4) \tag{21}$$

Because the result of adding two Taylor expansions

$$u(x+\Delta x,y-\Delta y) + u(x-\Delta x,y+\Delta y) = 2u(x,y) + \frac{\partial^2 u(x,y)}{\partial x^2} \Delta x^2$$

$$+ \frac{\partial^2 u(x,y)}{\partial y^2} \Delta y^2 - 2\frac{\partial^2 u(x,y)}{\partial x \partial y} \Delta x \Delta y + 0\ (\Delta x^4 + \Delta y^4) \tag{22}$$

Eq. 21 can be simplified by substituiting Eq. 22 into Eq. 21.

$$\frac{\partial^2 u(x,y)}{\partial x \partial y} = \frac{1}{4\Delta x \Delta y} \Big[u(x+\Delta x,y+\Delta y) + u(x-\Delta x,y-\Delta y)$$

$$- u(x+\Delta x,y-\Delta y) - u(x-\Delta x,y+\Delta y) \Big] + 0\ (\Delta x^2 + \Delta y^2) \tag{23a}$$

Eqs. 20a, 20b and 23a are refered to as central differencing. The first derivatives can also be expressed by central differencing technique by combining Eqs. 16 and 18.

$$\frac{\partial u(x,y)}{\partial x} = \frac{u(x+\Delta x,y) - u(x-\Delta x,y)}{2\Delta x} + 0\ (\Delta x^2) \tag{24a}$$

$$\frac{\partial u(x,y)}{\partial y} = \frac{u(x,y+\Delta y) - u(x,y-\Delta y)}{2\Delta y} + 0\ (\Delta y^2) \tag{24b}$$

When executing numerical computations, it is necessary to define a notation such that the individual mesh points are indiced. The lattice of mesh points used in this work is shown in Fig. 9.

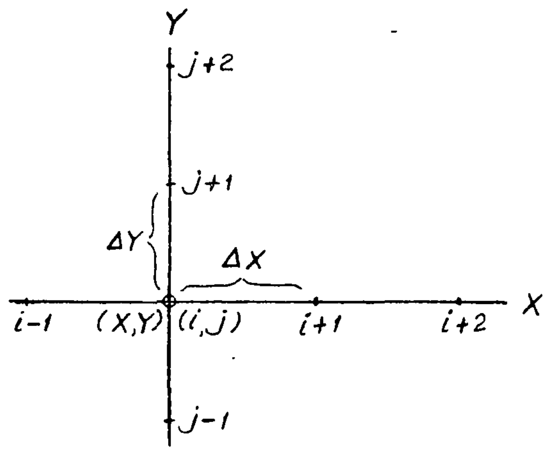


Fig. 9.

Lattice of mesh points

The finite difference approximation of derivatives, Eqs. 17a, b, 19a, b, 20a, b, 23a and 24a, b yield the following expressions

$$(\partial u/\partial x)_{i,j} = [u(i+1,j) - u(i,j)]/\delta x \tag{17c}$$

$$(\partial u/\partial y)_{i,j} = [u(i,j+1) - u(i,j)]/\delta y \tag{17d}$$

$$(\partial u/\partial x)_{i,j} = [u(i,j) - u(i-1,j)]/\delta x \tag{19c}$$

$$(\partial u/\partial y)_{i,j} = [u(i,j) - u(i,j-1)]/\delta y \tag{19d}$$

$$(\partial^2 u/\partial x^2)_{i,j} = [u(i+1,j) - 2u(i,j) + u(i-1,j)]/\delta x^2 \tag{20c}$$

$$(\partial^2 u/\partial y^2)_{i,j} = [u(i,j+1) - 2u(i,j) + u(i,j-1)]/\delta y^2 \tag{20d}$$

$$(\partial^2 u/\partial x\partial y)_{i,j} = [u(i+1,j+1) + u(i-1,j-1) - u(i+1,j-1) - u(i-1,j+1)]/4\delta x\delta y \tag{23b}$$

$$(\partial u/\partial x)_{i,j} = [u(i+1,j) - u(i-1,j)]/2\delta x \tag{24c}$$

$$(\partial u/\partial y)_{i,j} = [u(i,j+1) - u(i,j-1)]/2\delta y \tag{24d}$$

The indices will be written in the form of subscripts in the following discussion.

III.C Partial Donor Cell Formula

Following the conventional layout of variables and indices as shown in Fig. 10, there are at least three types of expressions that can be used to represent the momentum convective flux in the finite difference method (27).

Centered: $u_i = (u_{i-\frac{1}{2}} + u_{i+\frac{1}{2}})/2$

ZIP: $u_i = (u_{i-\frac{1}{2}}\ u_{i+\frac{1}{2}})^{\frac{1}{2}}$

Donor: $u_i = [(1+\bar{\alpha})\ u_{i-\frac{1}{2}} + (1-\bar{\alpha})u_{i+\frac{1}{2}}]/2$

where $\bar{\alpha} = \pm 1$ Complete Donor

$1 > \bar{\alpha} > -1$ Partial Donor

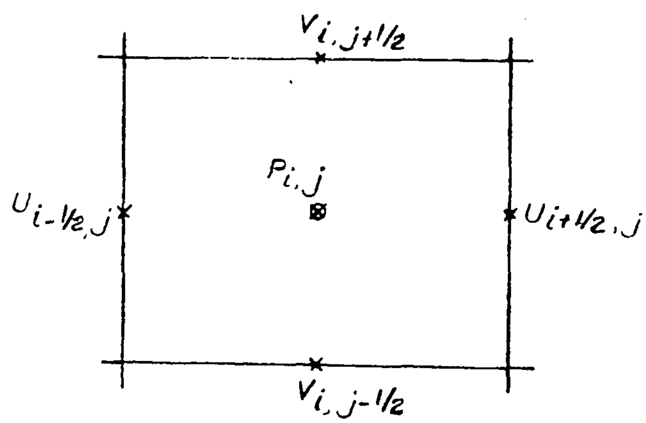


Fig. 10. Conventional layout of variables and indices in the mesh.

The centered difference type has less truncation error, compared with the donor difference type, but has tendency to be numerically unstable. The ZIP technique is more accurate than the donor difference type and has better stability than the centered difference method. The donor approximation has the best stability, but the magnitude can be excessive and cause error, e.g., for flow at high Reynolds numbers. The transient fluid motion occuring in rotational molding has two characteristics.

First, it's a low Reynolds number flow and, secondly, it starts with a violent initiation in the early stage. This becomes a special circumstance to suit the donor cell flux. The partial donor cell formula providing a flexible combination of stability and accuracy is the choice for approximating the derivatives.

III.D <u>Difference Equations</u> (28)

The cross-section of cylinder is divided into rectangular cells of width δx and height δy (Fig. 11). The fictitious boundary cells (shaded area) are added to each side of the region to facilitate the computation. Since the fractional indices as shown in Fig. 10 are not allowed in a Fortran program, the indexing in Fig. 11 is written in the actual code. The density is assumed constant and, therefore, is lumped into the gravitatational and viscosity terms. The pressure term is placed at the cell center. The velocity u(i,j) is in the middle of right side of the cell, and v(i,j) is at the middle of the top side, Fig. 12.

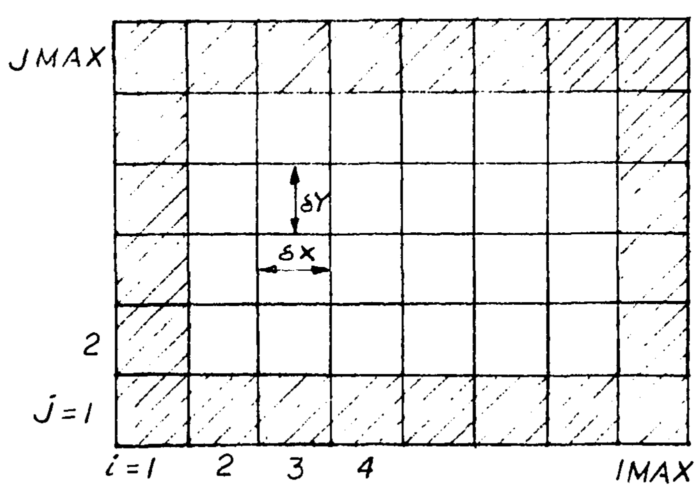


Fig. 11. Mesh indexing.

By the partial donor cell differencing method, Eqs. 14a, b and 15 are approximated by the finite difference forms:

$$u^o_{i,j} = u_{i,j} - \delta t \left[(p_{i+1,j} - p_{i,j})/\delta x - g_x + UUX + UVY - VISU \right] \quad (25a)$$

$$v^o_{i,j} = v_{i,j} - \delta t \left[(p_{i,j+1} - p_{i,j})/\delta y - g_y + UVX + VVY - VISV \right] \quad (25b)$$

$$(u^o_{i,j} - v^o_{i-1,j})/\delta x + (v^o_{i,j} - v^o_{i,j-1})/\delta y = 0 \quad (26)$$

where the momentum convective and viscous fluxes are approximated by

$$UUX = \partial uu/\partial x = \left[(u_{i,j} + u_{i+1,j})^2 + \alpha |u_{i,j} + u_{i+1,j}| (u_{i,j} - u_{i+1,j}) - (u_{i-1,j} + u_{i,j})^2 - \alpha |u_{i-1,j} + u_{i,j}| (u_{i-1,j} - u_{i,j}) \right] / 4\delta x \quad (27a)$$

$$UVY = \partial uv/\partial y = \left[(v_{i,j} + v_{i+1,j})(u_{i,j} + u_{i,j+1}) + \alpha |v_{i,j} + v_{i+1,j}| (u_{i,j} - u_{i,j+1}) - (v_{i,j-1} + v_{i+1,j-1}) \times (u_{i,j-1} + u_{i,j}) - \alpha |v_{i,j-1} + v_{i+1,j-1}| (u_{i,j-1} - u_{i,j}) \right] / 4\delta y \quad (27b)$$

$$VISU = \nu (\partial^2 u/\partial x^2 + \partial^2 u/\partial y^2) = \nu \left[(u_{i+1,j} - 2u_{i,j} + u_{i-1,j})/\delta x^2 + (u_{i,j+1} - 2u_{i,j} + u_{i,j-1})/\delta y^2 \right] \quad (27c)$$

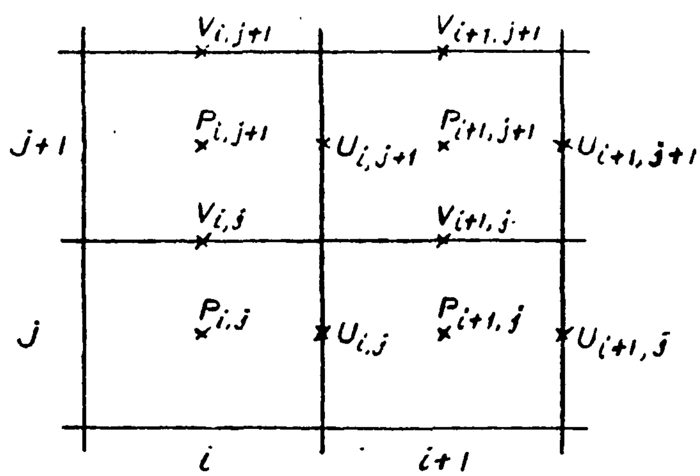


Fig. 12. Locations of cell variables in a mesh.

$$UVX = \partial uv/\partial x = [(u_{i,j} + u_{i,j+1})(v_{i,j} + v_{i+1,j}) + \alpha|u_{i,j} + u_{i,j+1}|(v_{i,j} - v_{i+1,j}) - (u_{i-1,j} + u_{i-1,j+1}) \times (v_{i-1,j} + v_{i,j}) - \alpha|u_{i-1,j} + u_{i-1,j+1}|(v_{i-1,j} - v_{i,j})]/4\delta x \quad (27d)$$

$$VVY = \partial vv/\partial y = [(v_{i,j} + v_{i,j+1})^2 + \alpha|v_{i,j} + v_{i,j+1}| \times (v_{i,j} - v_{i,j+1}) - (v_{i,j-1} + v_{i,j})^2 - \alpha|v_{i,j-1} + v_{i,j}| \times (v_{i,j-1} - v_{i,j})]/4\delta y \quad (27e)$$

$$VISV = \nu(\partial^2 v/\partial x^2 + \partial^2 v/\partial y^2) = \nu[(v_{i+1,j} - 2v_{i,j} + v_{i-1,j})/\delta x^2 + (v_{i,j+1} - 2v_{i,j} + v_{i,j-1})/\delta y^2] \quad (27f)$$

The superscripts in the equations denote a new time level, i.e., at time $(n+1)\delta t$. The terms with no superscript are at the old time level, i.e., at time $n\delta t$. The coefficient α has a value from 0 to 1. The details of deriving Eqs 25a, b and 26 are included in Appendix I.

IV COMPUTER MODEL

The numerical model is based on SOLA-SURF algorithm (28), which is a simplified Marker-and-Cell (MAC) method (29). The MAC method was the first technique to use pressure and velocity as the primary variables to treat problems involving free surface. The method defines the field variables in an Eulerian mesh of calculational cells as shown in Fig. 12. The method uses weightless marker particles to specify the free surface. The pressure boundary condition is applied at the cell centers. In the later versions of the MAC method, the crude approximation of pressure is improved by satisfying the free surface stress conditions more accurately and in applying the pressure at the actual locations instead of cell centers (30). The marker particles can also be eliminated. The SOLA-SURF code includes both improvements. However, it is not directly applicable to the rotational molding geometry because of the following reasons:

1. The code is written in Eulerian concept of a fluid moving through a stationary network of cells, while a rotational cylinder may be better represented by a Lagrangian system, in which the fluid is covered by a mesh of cells whose vertices move with the fluid. Since the Eulerian technique is most useful for problems involving large fluid distortion, this study will modify the SOLA-SURF code, but keep this advantage for a moving mesh of cells.
2. The code is for uni-directional flow. In a rotational cylinder, the fluid moves along the wall. The flow field will be treated as the combination of two uni-directional flows.
3. The code is for a confined free surface with no advancing front, but on the contrary, the present problem is characterized by the

expansion of fluid covered area. Initial conditions have to be defined for each newly entered call.

4. Although cylindrical coordinates can be used in this code and makes the mathematics easier for the cells in the cylinder, it is not suitable physically for this transient problem. The cell size in cylindrical coordinate system varies, smaller for those cells near the center of the cylinder and lager for those cells near the wall. This is advantageous to the calculation of the free surface because of smaller cells. But the larger cell area is where the advancing front of fluid is entering. The large cell can not meet the requirement of high accuracy. Therefore, a Cartesian coordinates is used and, consequently, it is necessary to define the curve boundary.

The major considerations of modeling the Eqs. 25a, b and 26, including iteration procedure, initial conditions, boundary conditions and numerical stability, are discussed in the following sections.

IV.A Rigid Surface Location and Slope

The cell size is uniform throughout the whole cross-section area of the cylinder. The cells at a rigid boundary are partially inside the cylinder and partially outside. A criterion is necessary to determine if a specific boundary cell is "inside" the cylinder. When a boundary cell is considered "outside" the cylinder, it becomes the fictitious cell as shown in Fig. 11. In this calculation, if the inside fraction of the total cell area is greater than or equal to ¼, it is flagged "inside" (31). To calculate the inside area, a straight line (dotted line) connecting intersections of curve and Y-division lines is used in place on the curve boundary as shown in Fig. 13. The area of trapezoid GHED approximates the bottom cell area at X = i + 1, and FGDC the bottom cell area at X = i. Both have an area greater than a quarter of the uniform cell area $\delta x \delta y$. Hence, both cells (i,j) and (i + 1,j) are considered "inside" the boundary. The BFC area is smaller than $\delta x \delta y/4$ and, therefore, the cell (i - 1,j) is not inside the boundary. The bottom cell at X = i - 1 can be found to be (i - 1, j + 1).

For a curved boundary, the fictitious cells are not outside a rectangle as shown in Fig. 11. Some of the fictitious cells may be at j = 1. Some may be at j = 2, 3, 4, etc. Likewise, they are not limited to i = 1 and IMAX. But every fictitious cell is, by definition, immediately next to the boundary cell.

At each boundary cell, the velocity of the boundary surface is known. This velocity vector can be viewed as a combination of the two Cartesian velocity components u and v, which can be calculated from the

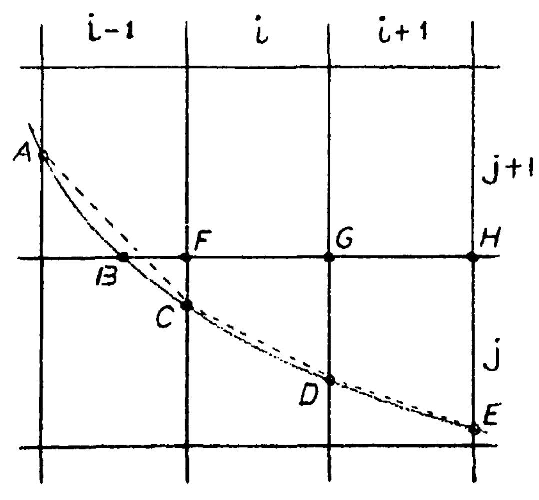


Fig. 13. Approximation of curved boundary.

equations.

$$BU_i = AS \cos (\tan^{-1} SLOB_i)$$

$$BV_i = AS \sin (\tan^{-1} SLOB_i)$$

where BU = u velocity at boundary, cm/sec

BV = v velocity at boundary, cm/sec

AS = Rotating velocity, cm/sec

SLOB = Tangential slope at boundary

The SOLA-SURF algorithm is restricted to the cells with slopes of the surface equal to or less than the aspect ratio of cell ($\delta y/\delta x$). If the slope is greater than the aspect ratio, the simple boundary conditions for the free surface in sec. IV.F.1 will not be applicable. The simple forms for approximating the free surface location in sec. IV.D will involve too much error. In a cylinder, it is predictable that the slope of free surface will increase rapidly as the cells approach either the left or right side boundary. This may be due to the steep slope of the rigid surface at both side boundaries. Therefore, an i column is excluded from the "inside" of cylinder if its rigid surface has a slope greater than $\delta y/\delta x$. This exclusion should not distort the general picture of cylinder. A 5% or less exclusion of the total area would be acceptable. Both the calculations of surface location and slope are outlined in Appendix II.

IV.B <u>Sign Convention</u>

The cylinder is divided into two parts: lower and upper halves. Since the i-indexing is following the direction of rotation, the indexing is from left to right for the lower half of the cylinder and right to left for the upper half of the cylinder if the rotation is counter-clockwise. U-velocity is positive (+) when it is following the indexing direction and negative (-) when counter to the direction. Th j-indexing is from bottom to the top. Both halves have the same indices. V-velocity is positive when it is upward and negative when it is downward. The gravitational constants g_x and g_y also have the same convention as u- and v-velocities. Hence, g_y being downward for all of the time has a negative sign. Fig. 14a shows these conventions. When a cell is involved in calculations of both halves, three points should be noted. First, it will have two i-indices, one for the lower half and another for the upper. Second, the u-velocity will have different signs: namely, positive for the lower half and negative for the upper, or vice verse. Third, the placing of u-velocity in the upper half is different from in the lower half because of the difference in rotation directions as shown in Fig. 14b.

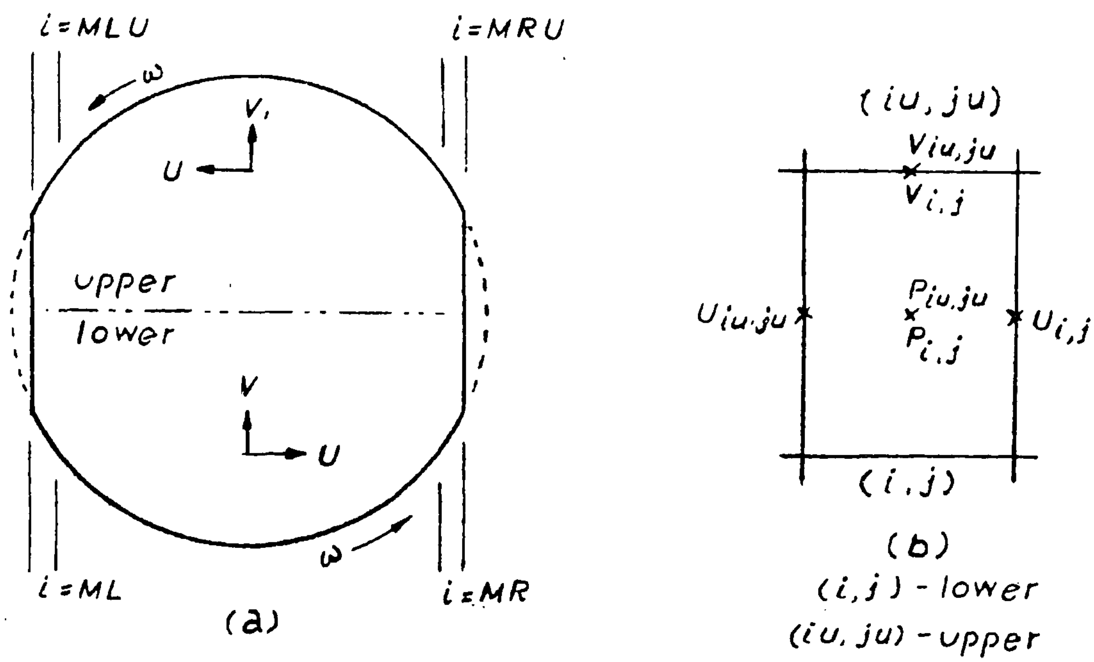


Fig. 14. Sign Convention.

IV.C Velocity Iteration

The velocity iteration procedure in SOLA-SURF is adopted in this model. In every time cycle, the velocities of each cell is predicted from Eqs. 25a, b. These predictions probably will not satisfy Eq. 26. The discrepancy is defined by

$$D = (u_{i,j} - u_{i-1,j})/\delta x + (v_{i,j} - v_{i,j-1})/\delta y \tag{28}$$

When absolute value of D is less than a specified value called tolerance, the iteration is terminated for that time cycle. Otherwise, the pressure is adjusted proportional to the discripancy.

$$\delta p = -D/(2\delta t/\delta x^2 + 2\delta t/\delta y^2) \tag{29}$$

A new set of velocities is calculated according to the following equations:

$$u_{i,j} = u_{i,j} + \delta p \delta t/\delta x \tag{30a}$$

$$u_{i-1,j} = u_{i-1,j} - \delta p \delta t/\delta x \tag{30b}$$

$$v_{i,j} = v_{i,j} + \delta p \delta t/\delta y \tag{30c}$$

$$v_{i,j-1} = v_{i,j-1} - \delta p \delta t/\delta y \tag{30d}$$

Returning to Eq. 28 for convergency check, these new predictions start a new iteration. This procedure repeats until $D \leq$ tolerance is satisfied in every cell of the network. A new time cycle will, then, start. The derivations of Eqs. 29 and 30a - d are described in Apprendix III.

It should be noted that Eq. 29 is applicable only to cells at interior mesh points. For cells at the free surface or a rigid boundary, the following equations are used:

Top Free Surface

$$\delta p = (1 - \eta_T)p_{i,jT-1} - p_{i,jT} \tag{31a}$$

where $\eta_T = \delta y/[H_i - (jT - 2.5)\delta y]$

jT = Y - index of top cell

H_i = height of top surface from mesh baseline

Bottom Free Surface

$$\delta p = (1 - \eta_B) p_{i,jB+1} - p_{i,jB} \quad (31b)$$

where $\eta_B = \delta y / [(jB - 0.5)\delta y - HB_i]$

jB = Y - index of bottom cell

HB_i = height of bottom surface from mesh baseline

Top Rigid Surface

$$\delta p = [v_{i,jT-1} - (u_{i,jT} + u_{i-1,jT})(H_{i+1} - HB_{i-1})/4\delta x - \lambda_T (u_{i,jT} - u_{i-1,jT})\delta y/\delta x]/[(1 + 2\lambda_T \delta y^2/\delta x^2)\delta t/\delta y] \quad (31c)$$

where $\lambda_T = (jT - 2) - H_i/\delta y$

Bottom Rigid Surface

$$\delta p = [v_{i,jB-1} - (u_{i,jB} + u_{i-1,jB})(H_{Bi+1} - HB_{i-1})/4\delta x - \lambda_B (u_{i,jB} - u_{i-1,jB})\delta y/\delta x]/[(1 + 2\lambda_B \delta y^2/\delta x^2)\delta t/\delta y] \quad (31d)$$

where $\lambda_B = (jB - 1) - HB_i/\delta y$

The derivations in details of Eqs. 31a - d are explained in Appendix IV.

IV.D Free Surface Position

After the velocity field converges in every cell, a time cycle is concluded with a set of new free surface positions. The movement of free surface can be calculated according to the kinematic equations, which involve velocities and surface slope. The velocities u(i,j) and v(i,j) for a particular cell are assumed applying to the center of the surface, O. The particle at O moves to an arbitrary point D during the time period t, as shown in Fig. 15(a). Paralleling to the orginial surface, a line CD is drawn as shown in Fig. 15(b). The net displacement of the free surface is from O to C. By a geometrical relationship,

$$OC = ED = AD - AE$$

$$= OB - OA\,(\Delta H/\delta x)$$

That is equivalent to

$$H_i = v\,\delta t - u\,\delta t(\Delta H/\delta x) \tag{32}$$

where u,v = mid-surface velocities

Eq. 32 is expressed in finite difference form to obtain the equation for top free surface.

$$H_i^o = H_i + \delta t\left\{-\left[(u_{i,jT} + u_{i-1,jT})\,(H_{i+1} - H_{i-1}) - \gamma\,|u_{i,jT} + u_{i-1,jT}|\,(H_{i+1} - 2H_i + H_{i-1})\right]/4\delta x + \lambda_T\, v_{i,jT} + (1 - \lambda_T)\, v_{i,jT-1}\right\} \tag{33a}$$

where γ = weight of partial donor cells

λ_T = ratio of liquid height to δy

Similarly, the equation for bottom free surface is

$$HB_i^o = HB_i + \delta t\left\{-\left[(u_{i,jB} + u_{i-1,jB})\,(HB_{i+1} - HB_{i-1}) - \gamma\,|u_{i,jB} + u_{i-1,jB}|\,(HB_{i+1} - 2HB_i + HB_{i-1})\right]/4\delta x + (1 - \lambda_B)\, v_{i,jB} + \lambda_B\, v_{i,jB-1}\right\} \tag{33b}$$

where λ_B = ratio of void height to δy

The derivation of Eqs. 33a, b are described in Appendix V.

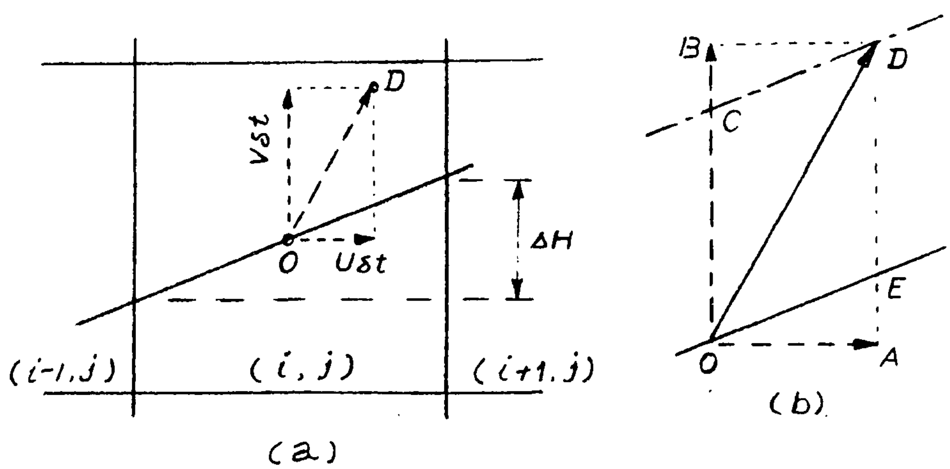


Fig. 15 Kinematic equation for free surface location

IV.E Initial Conditions

The initial conditions includes (1) the top free surface before the cylinder rotation is started, (2) the velocities and pressures at the moment the cylinder is starting to rotate, and (3) the velocities and pressures for the newly entered cells.

IV.E.1 Initial Top Free Surface

The location of the initial top free surface can be specified by either the most left or the most right contact point of the surface with the wall. For example, in Fig. 16(a), the distance of the most left point from i = 0 is

$$TX = AR - [AR^2 - (AR - FLHT)^2]^{1/2} \tag{34}$$

where AR = radius of cylinder, cm

FLHT = initial fluid height in the cylinder, cm

Since this algorithm is inapplicable to the i column of which the thickness of liquid layer is limited in one single cell, e.g., columns i_1 and i_2 in Fig. 16(b). The most left column for calculation, IL, may not be exactly at TX. IL is the first i column from the left with thickness of two or more cells.

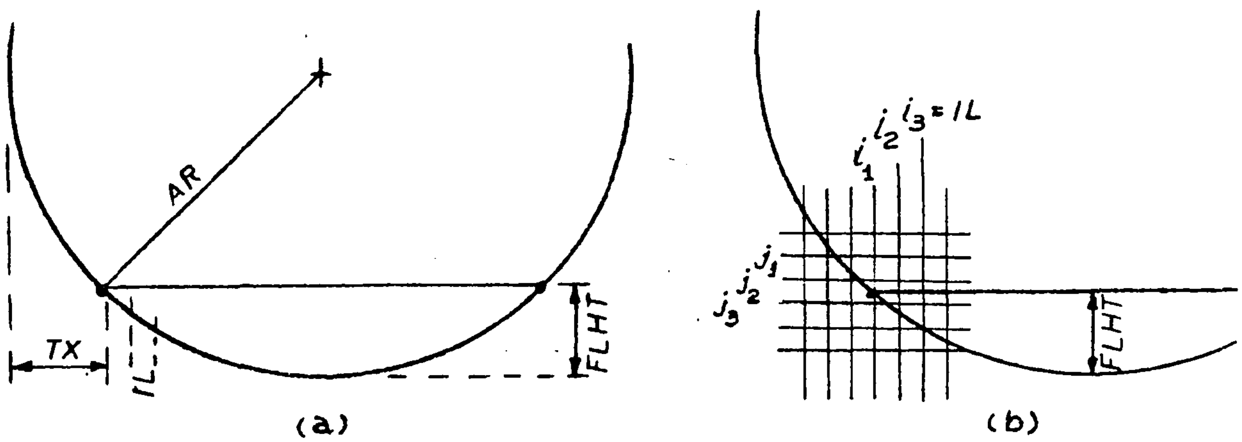


Fig. 16 Initial top free surface.

IV.E.2 Initial Velocities and Pressures

When the cylinder starts to rotate, it is assumed that only the cells at the boundary layer move. Accordingly,

$$u_{i,jB} = BU_i \tag{35a}$$

$$v_{i,jB} = BV_i \tag{35b}$$

The velocities at all other cells are assumed to be zero.

However, every cell filled with liquid completely or partially should have a pressure value. The initial pressure is the hydrostatic pressure.

$$p_{i,j} = - g_y \left[H_i - (j - 1.5)\delta y \right] \tag{35c}$$

IV.E.3 Initial Conditions for Newly Entered Cells

After the new free surface is calculated by Eq. 33b for the upper half, velocities and pressure have to be assumed for these newly entered cells so that the new velocities can be predicted from Eqs. 25a, b in the next time cycle.

$$u_{i,j} = (u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1})/4 \tag{37a}$$

$$v_{i,j} - (v_{i+1,j} + v_{i-1,j} + v_{i,j+1} + v_{i,j-1})/4 \tag{37b}$$

$$p_{i,j} = g_y \left[(j - 1.5)\delta y - HB_i \right] \tag{38}$$

When the calculation is in the lower half, these initial conditions are optional. The velocity iteration is able to converge with the assumption of zero pressures and zero velocities at the newly entered cells.

IV.E.4 Initial Conditions for Newly Entered Column

The assumed velocities in newly entered columns of cells are 2nd and 4th quardrants:

$$u_{i,j} = u_{i-1,j} \tag{39a}$$

$$v_{i,j} = v_{i-1,j} \tag{39b}$$

1st and 3rd quardrants:

$$u_{i,j} = 2u_{i-1,j1} - u_{i-2,j2} \tag{40a}$$

$$v_{i,j} = 2v_{i-1,j1} - v_{i-2,j2} \tag{40b}$$

where j1 and j2 have the same distance to the top rigid surface in i-1 and i-2 columns, respectively, as j in i column. Fig. 17 illustrates the relations among j, j1 and j2. The pressures in the new column are calculated by Eqs. 36 and 38.

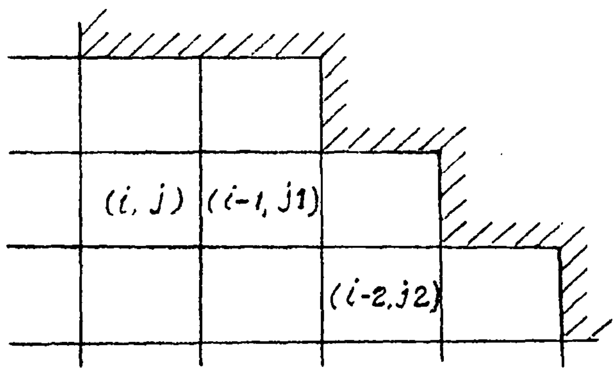


Fig. 17 Relative positions of j, j1 and j2.

IV.F <u>Boundary Conditions</u>

The boundary conditions include velocities at the boundary cells, the fictitious cells and the empty cells surrounding the free surface. A no-slip boundary condition can be applied to force the tangential velocities of the fluid at the walls to be identical to the rotational velocity at the walls. For a free surface, the continuative boundary condition will make the adjacent cell outside the fluid to have the same velocities as the cell inside the fluid.

IV.F.1 <u>Free Surface</u>

During the propagation of fluid movement, the image cells (i-1,j) of the front wave are assigned the velocities following continuative boundary condition.

$$v_{i+1,j} = v_{i,j} \qquad (41a)$$

$$u_{i+1,j} = u_{i,j} \qquad (41a)$$

For both top and bottom free surfaces, the fluid rarely occupies a full surface cell. Thus, v velocities can be calculated from the continuity equation.

Top Free Surface

$$v_{i,j} = v_{i,j-1} - (u_{i,j} - u_{i-1,j})\delta y/\delta x \qquad (42a)$$

$$u_{i,j+1} = u_{i,j} \qquad (42b)$$

Bottom Free Surface

$$v_{i,j-1} = v_{i,j} + (u_{i,j} - u_{i-1,j})\delta y/\delta x \qquad (43a)$$

$$u_{i,j-1} = u_{i,j} \qquad (43b)$$

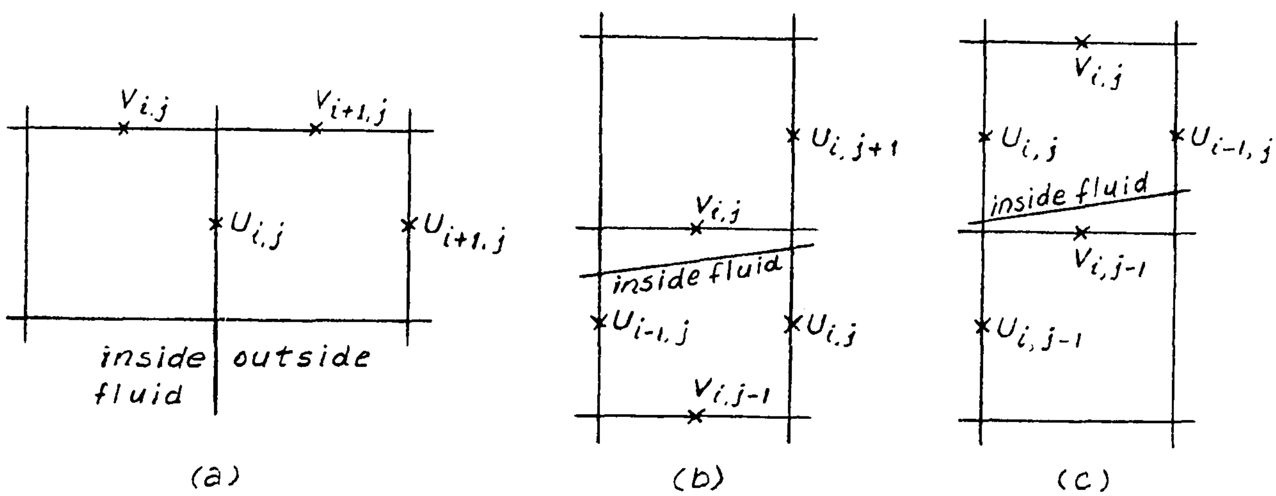


Fig. 18. Free surface boundary conditions (a) Eqs. 41a, b, (b) Eqs. 42a, b, (c) Eqs. 43a, b.

IV.F.2 Rigid Boundary

After excluding some cells from the calculation scheme as discussed in sec. IV.A, the cylinder is no longer complete circular, but a circular segment with two vertical flat portions on its right and left hand sides. It now becomes necessary to assume a rotating film occuping no space inside the wall and to make the cylinder stationary. This rotating film replaces the rotational cylinder. A Lagrangian system, thus, can be treated as a Eulerian system. Then, the horizontal velocities at the most right column for lower half and the most left column for upper half are zero. This forces the liquid to stop at the vertical walls. The vertical velocities may be assumed equal to the rotating speed of the cylinder.

Lower half:

$$u_{MR,j} = 0 \tag{44a}$$

$$v_{MR,j} = AS \tag{44b}$$

Upper half:

$$u_{MLU,j} = 0 \tag{45a}$$

$$v_{MLU,j} = -AS \tag{45b}$$

where MR = The most right column in the lower half

MLU = The most left column in the upper half

However, a better approach to obtain the v-velocity is by assuming a linear relation in the near wall area, as shown in Fig. 19(a).

$$v_{MR,j} = (2AS + v_{MR-1,j})/3 \tag{44c}$$

$$v_{MLU,j} = (-2\ AS + v_{MLU-1,j})/3 \tag{45c}$$

For cells outside the vertical walls, the following velocities are assumed:

$$u_{MR+1,j} = 0 \tag{46a}$$

$$v_{MR+1,j} = 2v_{MR,j} - v_{MR-1,j} \tag{46b}$$

$$u_{MLU+1,j} = 0 \tag{47a}$$

$$v_{MLU+1,j} = 2v_{MLU,j} - v_{MLU-1,j} \tag{47b}$$

For a bottom rigid boundary as shown in Fig. 19(b), $v_{i,j-1}$ is outside the wall. It can be calculated by equation of continuity, Eq. 43a. As the rotating velocity is the combination of u and v velocities,

$$AS = v_{i,j-1} \sin \theta + u_{i,j-1} \cos \theta$$

$u_{i,j-1}$ is computed by

$$u_{i,j-1} = (AS - v_{i,j-1} \sin \theta) \cos \theta \tag{48}$$

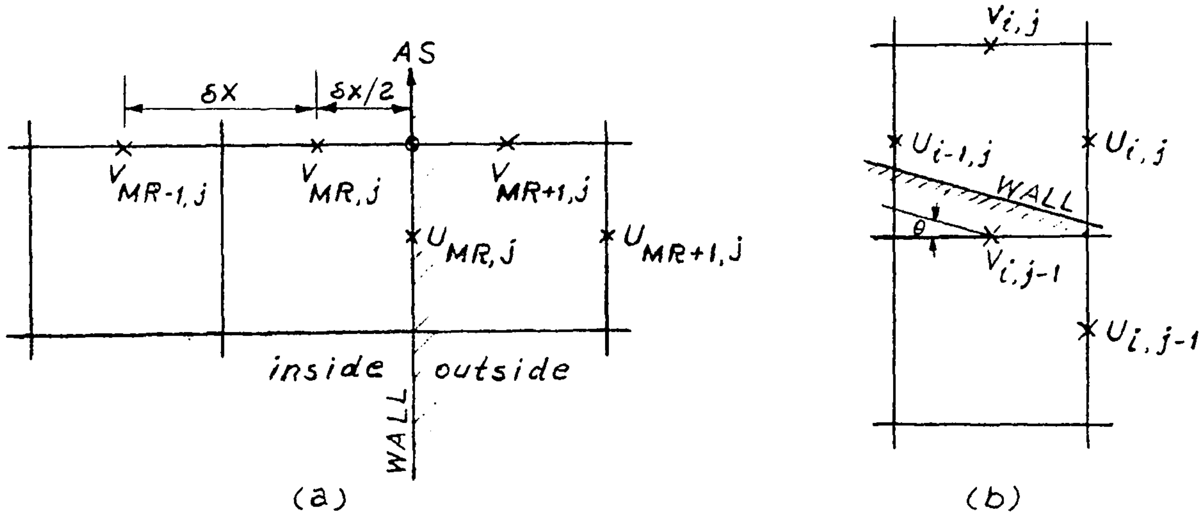


Fig. 19. Rigid boundary conditions.

IV.G Parameters and Numerical Stability

To insure numerical stability some parameters must be correctly chosen, e.g., the weights of partial donor cell method. Though some rules of thumb have been suggested, an optimum value can be achieved only by trial and error.

IV.G.1 α,γ - Weight of Partial Donor Cell

In Eqs. 25a, b, α has value in the range of 0 to 1 as stated previously. It is limited by the inequality

$$1 \geq \alpha = k \max \left(|u\delta t/\delta x| \ , \ |v\delta t/\delta y| \right) \qquad k > 1 \tag{49}$$

A good choice of k is suggested 1.2 to 1.5 (28).

γ in Eqs. 33a and b is often equal to α. Its only constraint is

$$1 \geq \gamma \geq 0 \tag{50}$$

The example problem in section IV.I demonstrates the allowable difference between values of α and γ, with $\alpha = 0.9$ and $\gamma = 0.2$.

IV.G.2 OMG - Over-Relaxation Factor

To accelerate the convergence of velocity iteration, OMG is incorporated into Eq. 29.

$$\begin{aligned} p &= -(\text{OMG})\ (D)/2\delta t(1/\delta x^2 + 1/\delta y^2) \\ &= -(\text{BETA})\ (D) \end{aligned} \tag{51}$$

where $\text{BETA} = \text{OMG}/2\delta t(1/\delta x^2 + 1/\delta y^2)$

OMG is in the range of 1 to 2. If it is 2, the iteration will become unstable. A value of 1.8 is often the optimum.

IV.G.3 ϵ - Tolerance of Discrepancy

The tolerance used as the criterion of convergence test for Eq. 28 should be small enough to ensure satisfying the continuity equation. A value of 10^{-3} is typical for non-dimensional velocity in the order of 0.1 - 1. If the value is too small, the improvement of accuracy is minimal. But the computer time becomes excessive. For this study, a value up to 0.1% of the average velocity is adequate.

IV.G.4 δt - Time Increment

The time increment is the most important factor in stability. There are at least two criteria to be satisfied (28).

$$\delta t < \min (|\delta x/u|, |\delta y/v|) \tag{52a}$$

$$\delta t < \delta x^2 \delta y^2 / 2 \nu (\delta x^2 + \delta y^2) \tag{52b}$$

Another criterion suggests to include a parameter $(\varphi + 1)$ in denominator of inequality 52b.

$$\delta t < \delta x^2 \delta y^2 / 2(\varphi + 1) \nu (\delta x^2 + \delta y^2) \tag{52c}$$

where the value of φ is in the range of 0 - 2 (30).

IV.H Computation Procedures

The input data of the calculation defines the block boundary (IMAX, JMAX), cell size (DELX, DELY), initial liquid height (FLHT), rotation velocity (AS), kinematic viscosity (NU) and gravitational acceleration (GX, GY). It includes also the above mentioned parameters α, γ, OMG, ϵ, and time increment. Then, the actual rigid boundary, the mesh of cells and initial free surface are drawn. Both initial profiles of velocity and pressure are set up.

A section computation is followed to calculate the fluid moving from left of right in the lower half. When the fluid reaches the right boundary, it climbs up along the wall up to the top right boundary. This is the computation in "lower section I" shown in Fig. 20. The velocities and pressure of the cells at the front of the wave are re-defined for the "upper section". Thus, the computation in upper half starts. This section is for liquid moving from right to left with a bottom free surface.

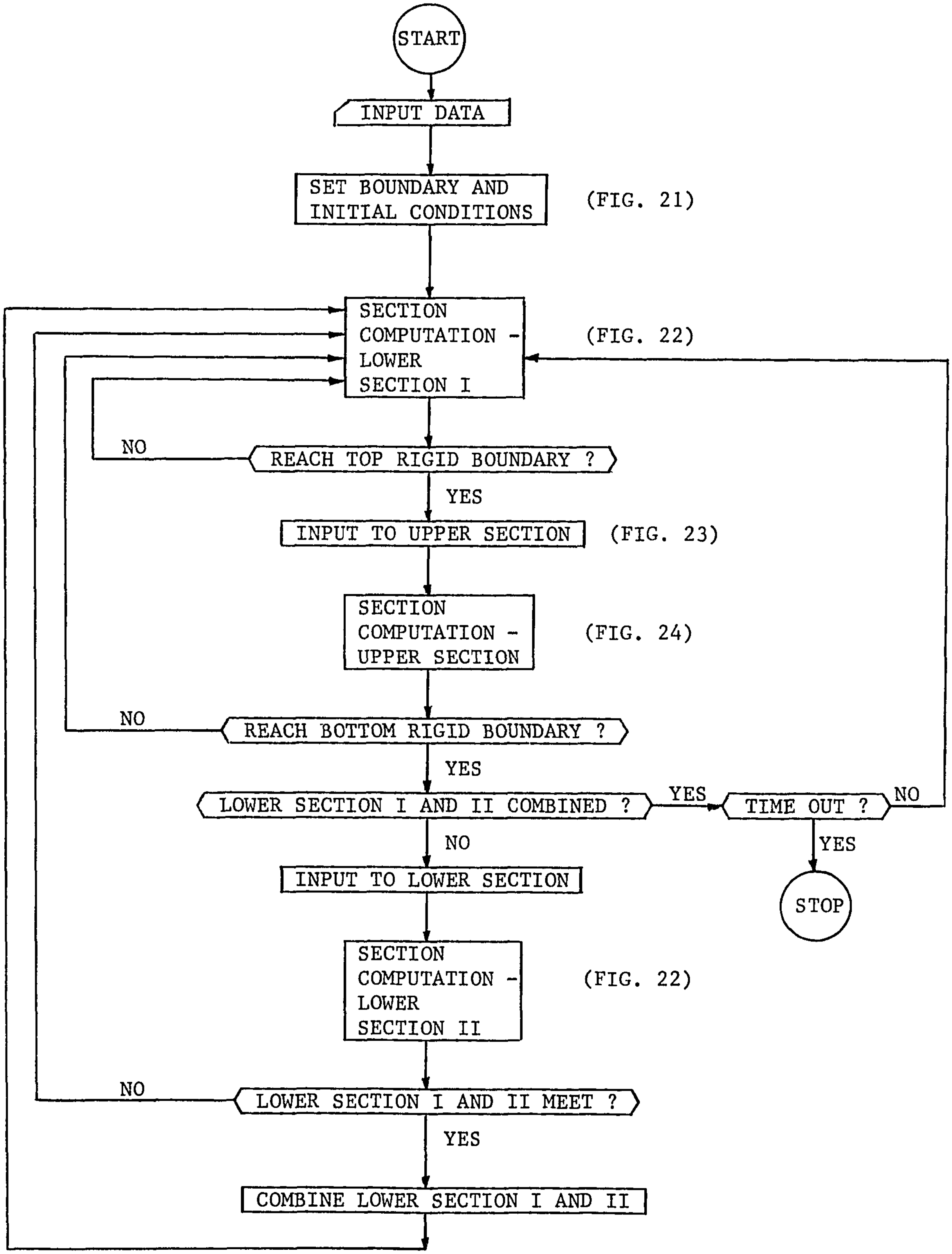


Fig. 20 Overall flow chart.

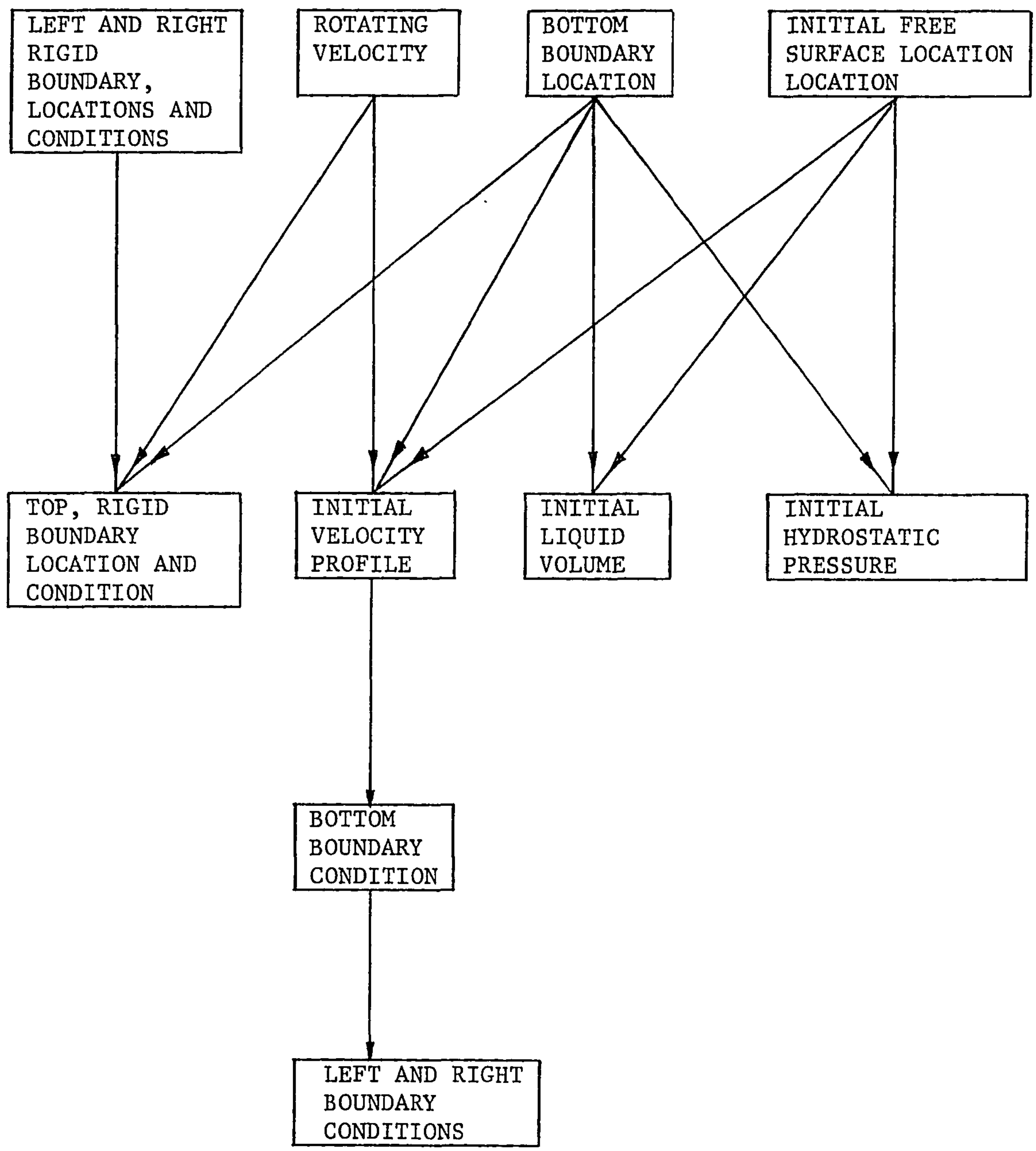


Fig. 21. Boundary and initial conditions

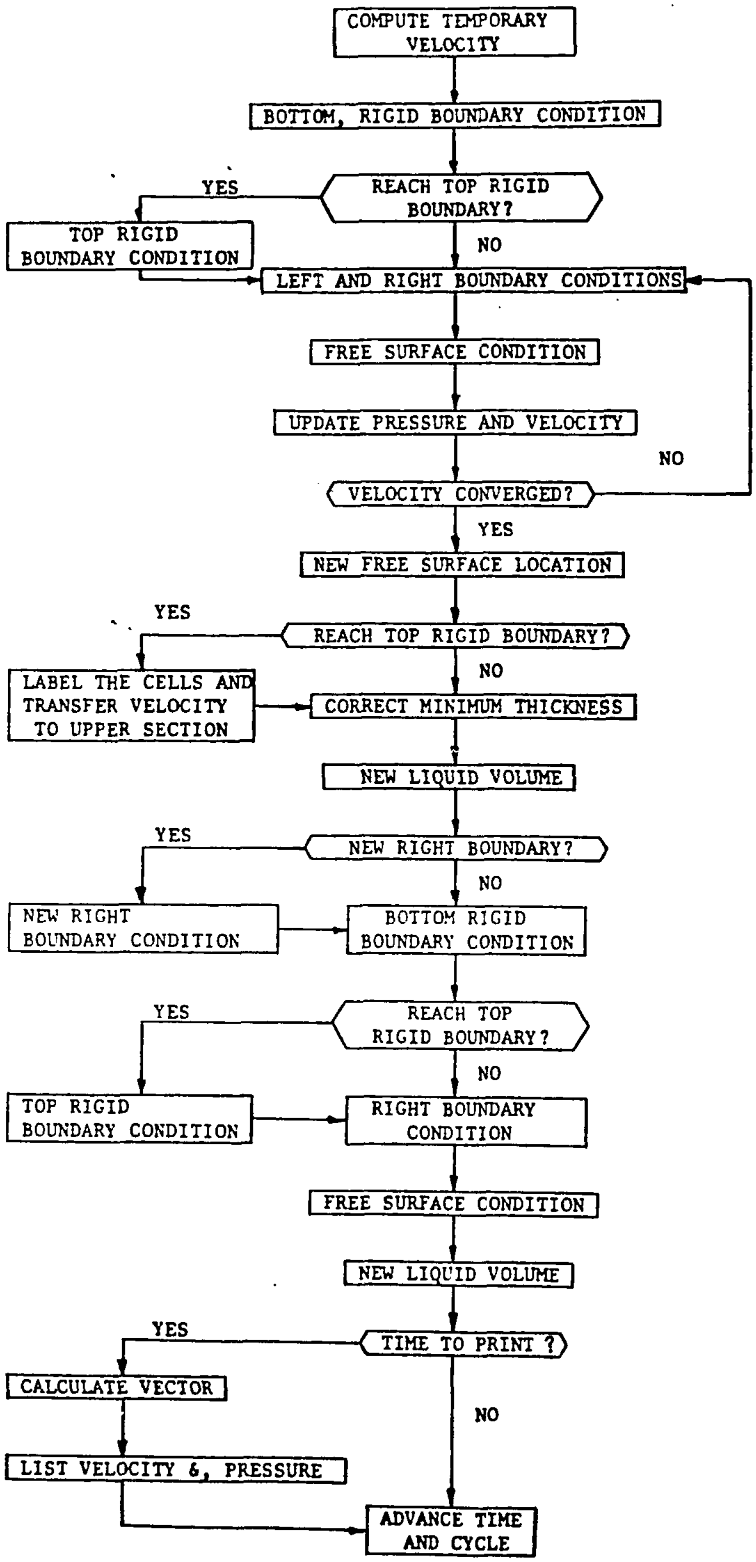


Fig. 22. Section computation - lower section I & II.

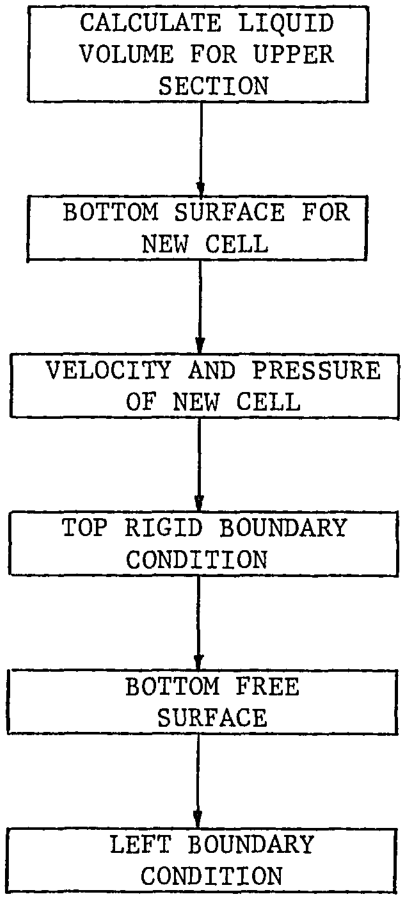


Fig. 23. Input to upper section.

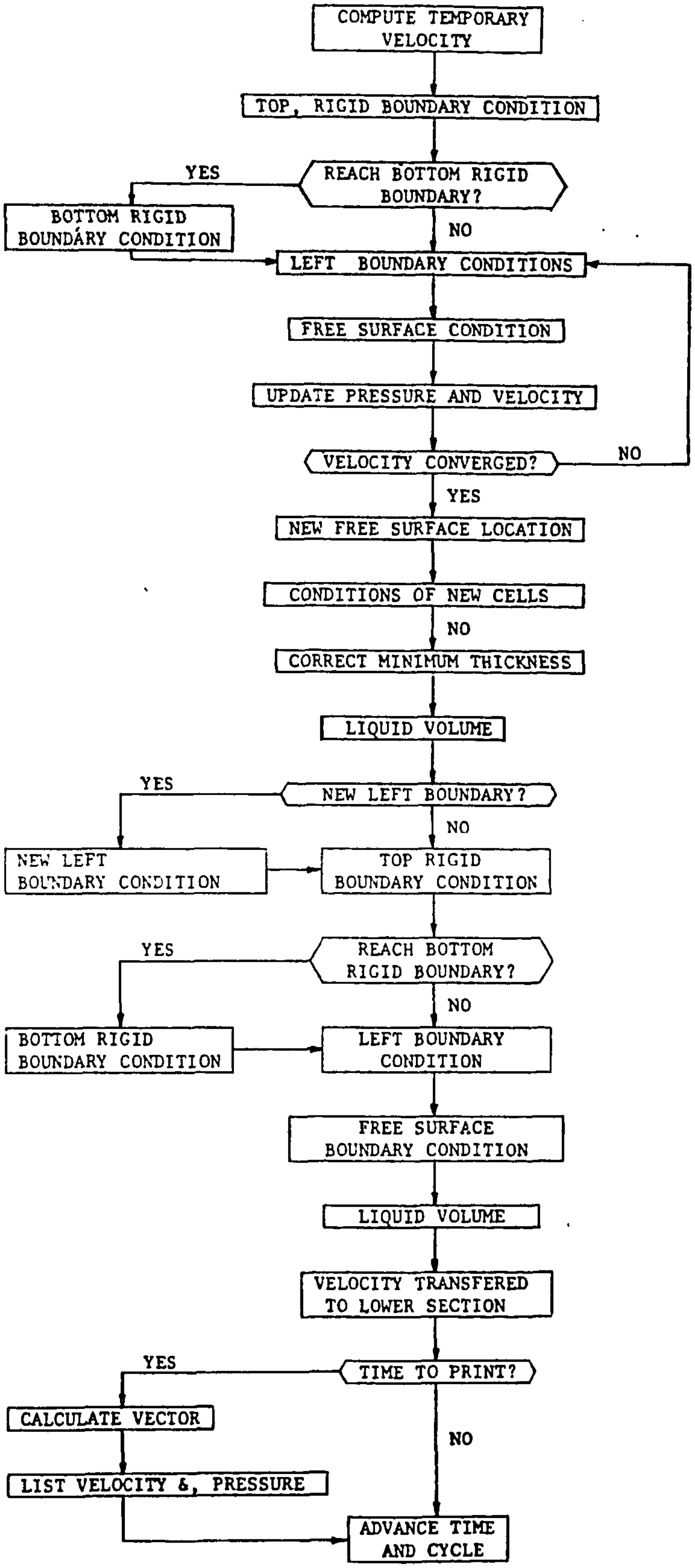


Fig. 24. Section computation - upper section.

These two section computations constitutes the calculation loop until the liquid reaches the left boundary and falls along the wall to the bottom rigid boundary. The computation of "lower section II" is added to the loop when the fluid touches the left bottom boundary. The front wave of lower section II will move faster to catch up the rear part of lower section I. After the liquid becomes a ring by connecting lower section I and II, the latter is consolidated into the former. The calculation loop consists of two sections -- lower I and upper -- again. But now the lower section I covers the entire area from ML to MR. The computational process will stop at specified time.

The section computations for lower and upper sections are basically the same. At the beginning the velocities are predicted by Eqs. 25a, b. Then, the boundary conditions for the rigid boundaries and free surfaces are set. If the discrepancy of continuity equation is less than tolerance, i.e., D (Eq. 28) $< \epsilon$, iteration of that time cycle is concluded. Otherwise, a new set of conditions are obtained from Eqs. 29 and 30 a - d. A test of convergency is repeated. When the velocities converge, a new free surface is located. Since the computation method requires minimum thickness of two cells in any column, it is necessary to eliminate the columns with single cell in the tail part of the fluid. On the other hand, a second cell is made to of each single-cell column in the remaining part. If the total liquid volume is less than the original, the fluid enters into a new column to keep a constant volume in the cylinder. The initial and boundary conditions of the new column are to be calculated before a new time cycle begins.

IV.I Example

An example is give with listing in Appendix VII. The cylinder size is chosen to be comparable to a laboratory equipment. The cell size is a compromise of accuracy and computer time. The kinematic viscosity was assumed to be in the same range as the silicone oil used in laboratory experiments. The input data are listed in Table I.

The indexing of the mesh is shown in Fig. 25. The horizontal index of lower half is 1-66, the upper 66-131. The vertical index is 1-34. After exclusion of cells of high aspect ratio, the mesh of calculation is reduced to 4-63 and 69-128, which includes one fictitious column of cells at each side. The excluded area is 3.4% of the circular cylinder. The actual circumference of the modified cylinder is 4.2% shorter than that of the whole circular cylinder. The initial liquid height results an average film thickness 11% of the radius, comparing with the maximum value of 20% in the rotational molding. Appendix VI details the calculations of these numbers

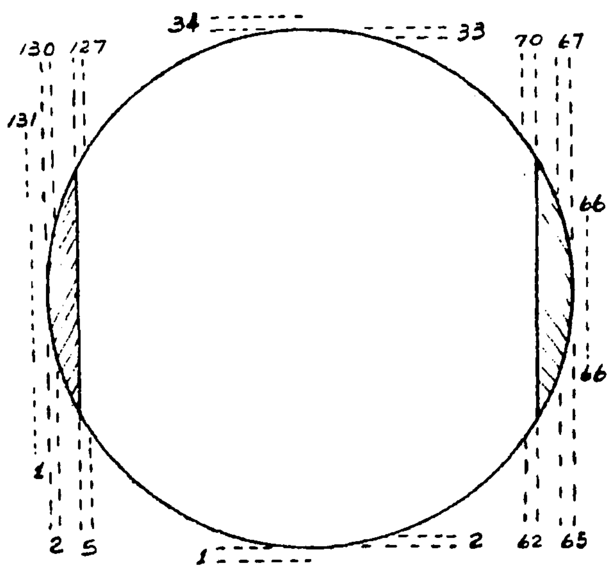


Fig. 25. Index in the example problem. Shaded area is excluded from calculation.

TABLE I
INPUT DATA OF EXAMPLE

ITEMS		VALUES
System		
Cylinder radius	(AR)	6.4 cm
Mesh spacing of cell X-axis	(δx)	0.2 cm
Mesh spacing of cell Y-axis	(δy)	0.4 cm
Rotation velocity	(AS)	220 cm/sec
Initial fluid height	(FLHT)	3.5 cm
Kinematic viscosity	(NU)	2 cm^2/sec
Gravitational acceleration X-axis	(g_x)	0
Gravitational acceleration Y-axis	(g_y)	-980 cm/sec^2
Parameters		
Weight of partial donor cell-convective flux	(α)	0.9
Weight of partial donor cell-free surface	(γ)	0.2
Over-relaxation factor	(OMG)	1.4
Convergence criterion	(ϵ)	0.1
Time increment	(δt)	0.0005 sec

NOTE: names of items in the listings

δx = DELX , δy = DELY

g_x = GX , g_y = GY

α = DPLHA (lower), ULPHA (upper)

OMG = OMG (lower) , UMG (upper)

γ = GAMMA

ϵ = EPSI (lower), UPSI (upper)

δt = DELT

The convergence criterion can be checked by assuming a value of discrepancy to be .01 for both u and v velocities in Eq. 28.

$$D = .01/.2 + .01/.4 = 0.75$$

Considering a rotational velocity of 220 cm/sec, the tolerance value of 0.1 represents error in the order of 10^{-3}%.

The time increment is to meet with both inequalities 52a, b.

$$\delta t < \min (|.2/220|, |.4/220|) = 0.0009$$

$$\delta t < (.2)^2 (.4)^2 /2(2) (.2^2 + .4^2) = 0.008$$

A value of 0.0005 is chosen by trial-and-error.

The controlling amount of donor cell fluxing has to satisfy the inequality 49.

$$1 \geq \alpha = k \max (|220 \times 0.0005/.2|, |220 \times 0.0005/.4|)$$

By trial-and-error method the k value is selected as 1.7 and α as 0.9. Similarly, the optimum values of γ and OMG are found to to be 0.2 and 1.4, respectively. Each parameter may have different values for lower and upper halves.

The result of calculation indicates the time of traveling around a cylinder is 327 time cycles. With the assumption of no-slip rotation throughout the calculation, the number of time cycles is 294 (Appendix VI). Since the actual number of time cycle is greater than the hypothetical number for no-slip rotation, the computer model is considered valid.

V DISCUSSION AND CONCLUSIONS

Several rotating velocities have been tested in the computer program. The comparison in Table II indicates correspondence between numerical analysis and experimental results which are summarized in Eqs. 1 and 2. By numerical method, the change from cascading to rimming flow occurs at a rotational velocity of 190–200 cm/sec. Eq. 2 gives the onset velocity of rimming flow at 203 cm/sec. Eq. 1 shows that the flow in the range of 140–190 cm/sec is cascading type. Fig. 26 shows the calculated points are

TABLE II

COMPARISON OF NUMERICAL METHOD WITH EXPERIMENTAL RESULTS

	AS (cm/sec)					
	140	180	190	200	210	220
t (sec)	.0008	.0006	.00055	.00055	.00053	.0005
cycle (1)	337	314	310	255	271	259
cycle (2)	390	335		258	277	260
cycle (3)	400	400	315	276	318	267
ILU (3)	122	122	121	127	127	127
flow	cascading ⟶		(4)	rimming ⟶		
w(rad/sec)	21.88	28.13	29.69	31.25	32.81	34.38
Re_m	5.36	6.89	7.27	7.66	8.04	8.42
$0.8\ Re_m^{0.867}$	3.43	4.27	4.47			
$1.11\ Re_m^{0.867}$				6.2	6.48	7.04
Fr_m	3.13	5.17	5.76	6.38	7.03	7.72

Notes: (1) This is the cycle number when liquid reaches I = 121 (upper)

(2) This is the cycle number when liquid reaches I = 122 (upper)

(3) This is the cycle number when liquid reaches I = ILU (upper)

(4) The trend of results in cycles 310 to 315 shows it will not become the rimming flow.

not in the experimental range, but fit the correlation of experimental results very well. The assumption of Newtonian fluid in the low shear

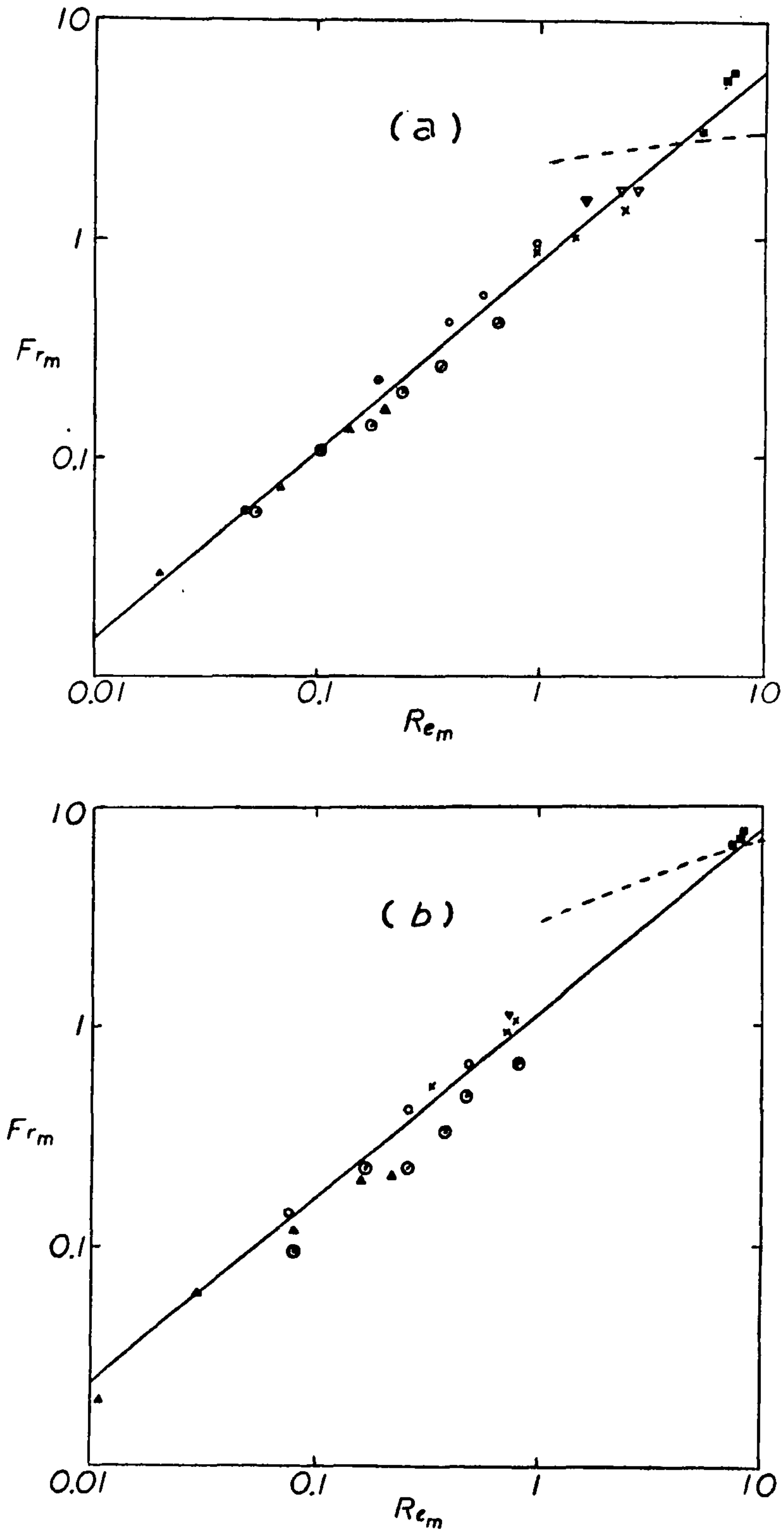


Fig. 26 Comparison of numerical method with experimental results (8). (a) cascading regime, (b) rimming regime.

Symbols	Viscosity (ν)
△	62.24
⊙	42.46
o	29.46
×	15.56
▽	12.45
■	2.0

NOTES:

1. Numerical method in this work is for 2 cm^2/sec.
2. The experimental correlations are shown by straight solid lines.
3. The prediction by Deiber and Cerro (24) is shown by dotted line.

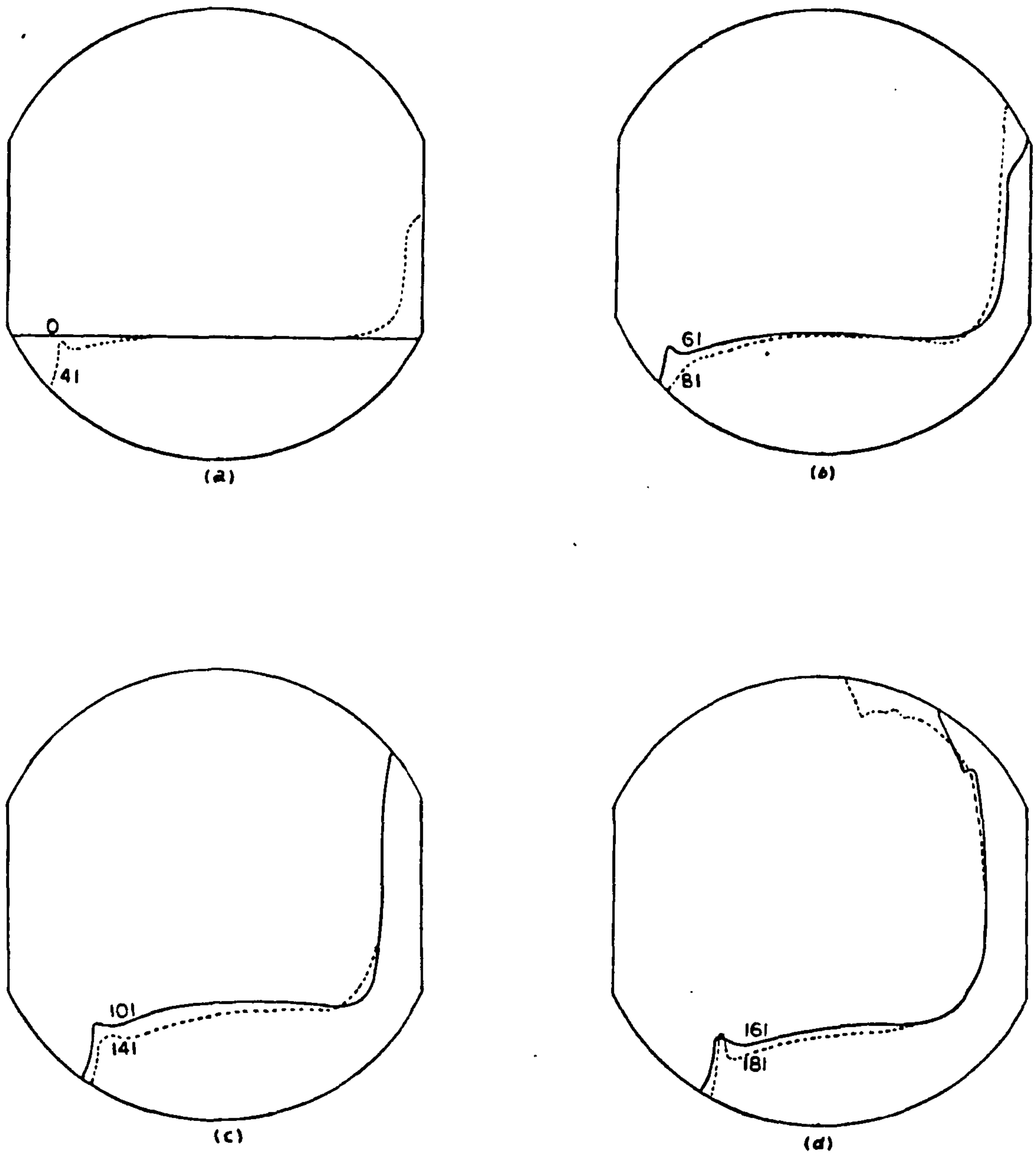


Fig. 27 Transient flow phenomena of the example. Free Surface changed as cylinder rotating from time cycle # 0 to 560.

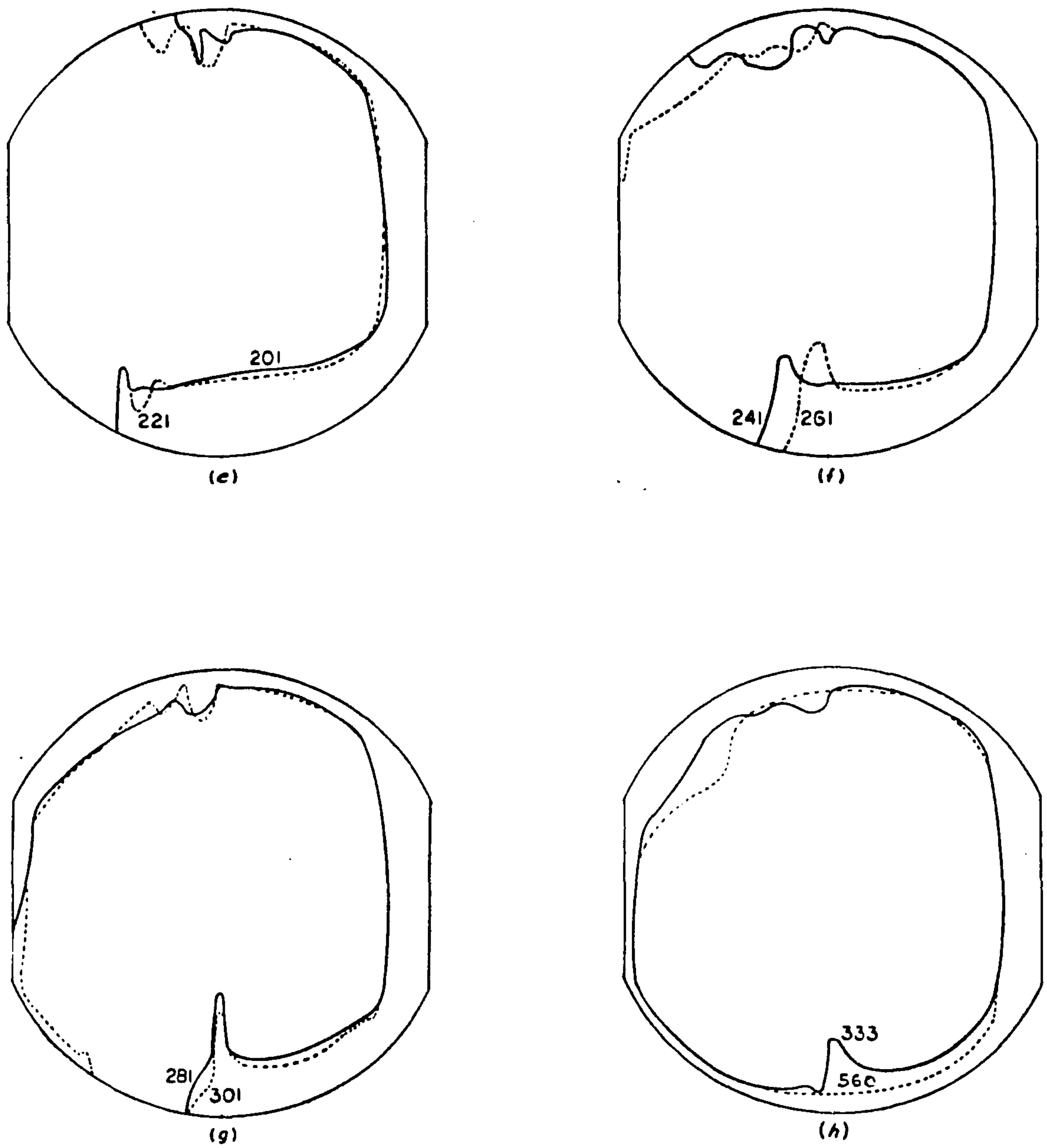


Fig. 27 Transient flow phenomena of the example. Free Surface changed as cylinder rotating from time cycle # 0 to 560. (cont'd)

rates and low Reynolds numbers (< 100) is also valid.

The transient flow phenomena of the example problem are shown in Fig. 27 (a) - (h). In Fig. 27 (a), a quiescent pool is shown at time cycle = 0, or prior to any motion. When the counter-clockwise rotation of the cylinder starts, a portion of the fluid is carried along the wall. In Fig. 27 (b), the fluid reaches the top rigid boundary at time cycle #61. One time cycle is equivalent to one t, or 0.0005 sec. As the time cycle advances from #61 to #261, the upper boundary is covered gradually by the fluid, while the liquid layer in the lower half becomes thinner and thinner. Coming down the vertical wall, the fluid moves fast to complete a ring at time cycle #333 as shown in Fig. 27 (h).

The accurate experimental data for comparison with the results of transient flow is still in need. The fluid velocities in the vicinity of rigid boundary are close to the rotating velocity. Backward velocities occur near the free surface. Both are expected. A comparison can also be made with steady rimming flow in previous works by Scriven et at. (21, 25). From Fig. 4, the thickest film should be at $\theta = 1.18$ for $Re_m = 7.66$. From Fig. 7a, which is for the value of $Re_m\ (R/b)^2$ equal to 1, the thinnest spot is $\theta = \pi$ and the thickest $\theta = 0$. In the physical dimension of the example problem, this is equivalent to $Re_m = 0.012$. Based on the series of change in Fig. 27 (a) - (h), the predicted thickest spot is in the range $\theta = 0$ to $\pi / 2$ and the thinnest $\theta = \pi$ to $3\pi / 2$. This concurs with the conclusion on page 12.

The results of calculation reveal variations of liquid volume with amount up to $\pm$ 10% as shown in Fig. 28. The average of variation is + 0.56%, and the standard deviation is 3.77%. This variation reflects the inaccuracy in approximating the position and pressure at the free surface. The tracking of the interface location is a surface height function method method, as described on page 33. This is a simple definition requiring a minimum storage of information. Other methods, such as line segments and marker particles, provide better approximation, but require more storage. The volume of fluid method (32), under testing for this rotational cylinder problem at present, has demonstrated very powerful in approximation and required only small amount of storage. The calculation of cell center pressure at the free surface on page 32 is merely an interpolation along y - direction. This is very simple, but may result incorrect pressure and, consequently, obtain spurious velocity. The appearance of humps at the free surface is a clear indication of this error. An improved method (30) is to use the nearest neighboring full cell for interpolation when there is more than one neighboring full cell. This will make the free surface smooth and may decrease the variation of volume.

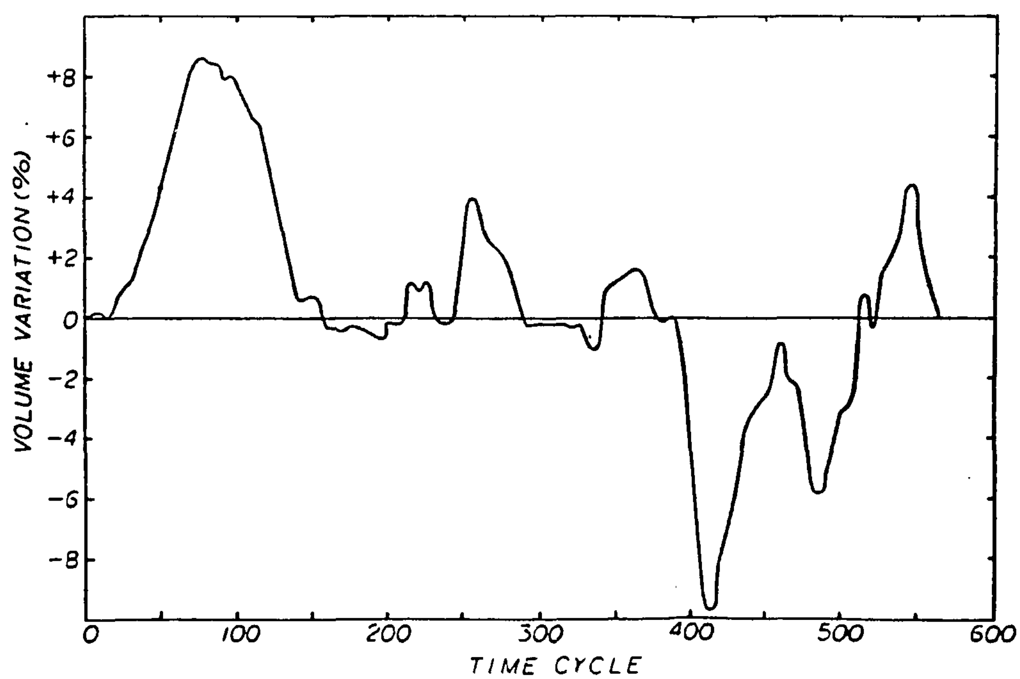


Fig. 28 Volume variation at time cycles.

In Fig. 27 (e) - (g), the tail part of the lower half results a high narrow wave. The bump stays in the tail and increases its height until a complete ring is formed. Then, it diminished rapidly. This is a phenomenon of singularity. It exists in all of the rotating velocities tested in this work. It builds up gradually as the fluid flow moves upwards and returns to a smooth surface in a later stage as shown in Fig. 29 (a). The number of time cycles to yield a smooth surface depends on the rotating velocity. The higher the velocity, the more the time cycle. This is illustrated in Fig. 29 (b).

Singularity is a result of discontinuity. In a field computation using finite difference technique, most of the discontinuities arise from the edges and noses of curvature. To overcome this difficulty, the mesh size may be reduced to a fine enough network (35). The singularity at

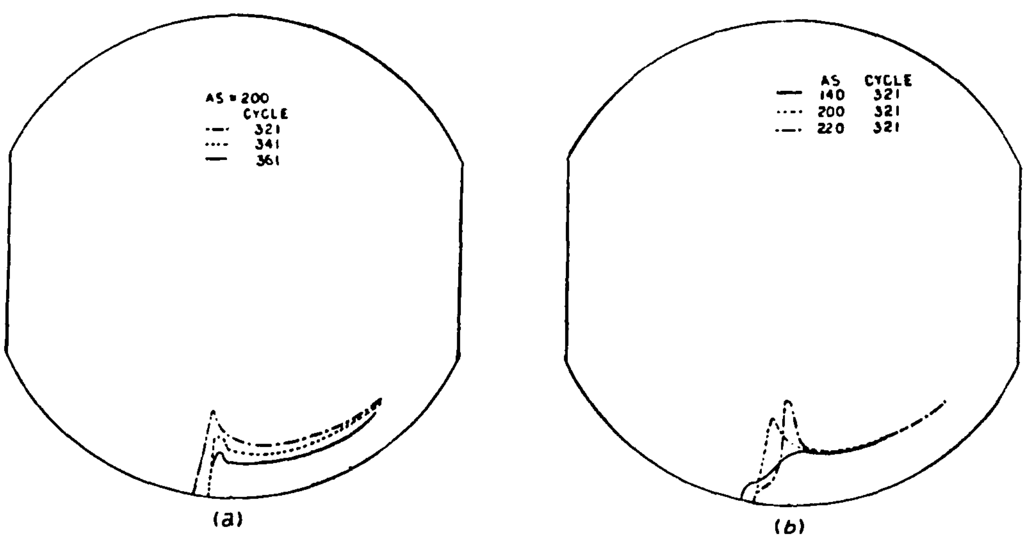


Fig. 29 Phenomena of singularity.

the contact line between solid and free surface has been investigated by Huh and Scriven (36) and others (37, 38). Huh and Scriven proposed a hydrodynamic model for steady state flow over a rigid surface. In terms of stream function, the solution for a flat, no-slip solid yielded a realistic velocity field. However, the shear stress and pressure field increased to infinity as the contact line was approached. The singularity indicates the breakdown of the hydro-dynamic model and suggests discontinuity physically existing around the contact line. The most important reason for this discontinuity is the no-slip boundary condition (53), that demands the liquid immediately next to the rigid wall have no relative velocity to the wall. According to no-slip condition the advancing contact angle should be a function of wall velocity as shown in Fig. 30. But experiments indicate the contact angle increases irregularly as the velocity increases. Both rolling and sliding have been mentioned in the observations. To correct the boundary condition, the no-slip condition may be replaced by a dynamic condition which includes a slip factor or momentum transfer coefficient varying in the range of no-slip to partial slip. Unfortunately, no such a condition has been proved successfully without certain limitations (39,

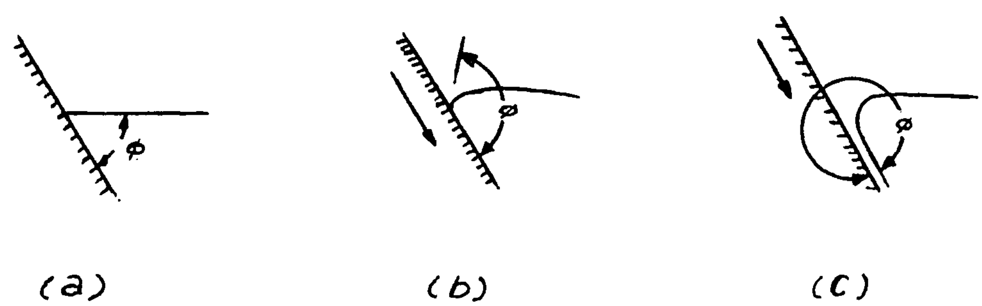


Fig. 30 Contact angle. (a) equilibrium, (b) low speed, (c) high speed.

40, 53). Other possible reasons for the singularity are cavitation, compressibility effects, breakdown of Newtonian fluid relation , etc. Due to the fact that surface tension has a significant effect on the characteristics of contact angle (54), it is impossible to obtain a complete picture of the development of singularity in any calculation excluding surface tension as in this work. Since the bump either diminishes in a later time cycle or exerts no significant effect on the result of rimming velocity, no attempt has been made to correct this difficulty.

During the development of model, various trials have been made to deduce some guidelines. The most important one, of course, is stability problem, which is inherent in all the numerical approximations. The set of parameters in the above example may not result in a convergence for other fluids of higher or lower veiscosity. In general, a set of parameters is good for a viscosity at several velocities. When the velocity changes, the main concern is time increment. Some test runs in the example indicate that a value of δt with 10% off its optimum may result in a divergence. Fortunately, the optimum ranges of parameters are not so narrow. Therefore, it is the parameters to be chosen first. If a set of parameters allows liquid to enter into the upper half, those values are adequate for the whole run.

The next important is the initial conditions in the upper half. Since the algorithm is using both the upstream and underneath cell velocities for computing discrepancy in Eq. 28, an intelligent guess of initial conditions is necessary to ensure a convergency in upper half. Eqs. 37a, b are found able to yield such a result. But other forms can be tried, too, for instance,

$$u_{i,j} = (u_{i-1,j} + u_{i,j+1})/2 \qquad (53a)$$

$$v_{i,j} = (v_{i-1,j} + v_{i,j+1})/2 \tag{53b}$$

An acceptable initial condition usually does not relate only to a single velocity of surroundings, such as Eqs. 39a, b. The initial conditions for a whole new column are less critical.

The third is correcting the minimum thickness. As mentioned in previous sections, this algorithm applies only to columns with at least two cells. If the free surface is inside the bottom cell of lower half or the top cell of the upper half, it becomes necessary to make up a second cell, i.e., to extend the liquid into the second cell. The question is how far it should extend to. In the lower half, it is to the top edge of the bottom cell.

$$H_i = (jB - 1)\,\delta y \tag{54}$$

In the upper half, it is to the bottom edge of the top cell.

$$HB_i = (jT - 2)\,\delta y \tag{55}$$

Comparison of this method with others is not available at present. (See Appendix VII for brief discussion on numerical methods for fluid dynamics.) The advantages of MAC method are basically its primitive variables and the capability to handle free surface conditions. The method has been proved extremely powerful in water wave problems. The example was run on a Amdahl 470 computer and required cpu 3 min 25.48 sec for 560 time cycles and 2 min 48.26 sec for 460 time cycles.

The accuracy of this method can be checked based on truncation and round-off errors. The truncation error in the centered differencing is second order in the spatial cell size and in the upstream differencing is first order. Since this method is using a hybird of centered and partial donor cell differencings, it can produce greater accuracy than the upstream differencing. The round-off error is minimized by double precision in computer calculation. Although cell refinement can alleviate these error problems, it is usually impractical for engineering purposes. When the difference equations are conditional stable, the fineness of time step must satisfy the stability criteria, such as inequalities 52a and b, that involve the cell size. Therefore, a decrease in cell size may cause δt decrease not by ratio of the later size to the previous size, but by square of the ratio. The computer time will increase rapidly to accommodate to the smaller cell size and much shorter time step. Other convective modeling techniques which have error of high order while avoiding the stability problems on a comparable large cell size is recommended for better accuracy. For instance, the Leonard algorithm (46) with error of third order could be tried. With this type of high efficiency technique, a cylindrical coordinate system with larger cell size can be considered so that the error in approximating the location of rigid boundary could be eliminated.

The question of multiple solution in a free-surface flow has been investigated primarily for flow between disks and swirling flow. Some steady state problems have been solved analytically or numerically to obtain non-unique solutions. At least three situations may lead to the multiple solutions:

(1) Wrong theory or assumption (41);

(2) Physically existing conditions, such as in the transition of couette flow and steady flow between rotating disks (42, 43); and

(3) The location and shape of free surface that is unknown a priori.

The last reason indicates the multiplicity of solution is inherent in the Navier-Stokes equation (44). Frequently, an approximation of the location and shape of free surface is built in the numerical method to provide assumed data of free surface. With those data, the iteration procedure will try to satisfy the free surface conditions. Due to the highly non-linear character of the Navier-Stokes equation, it is possible to have more than one set of location and shape satisfying the free surface conditions. Though it may be very difficult to prove uniqueness mathematically, only one of the solutins is expected physically possible, as discussed by Taylor (45). In MAC and MAC-derived techniques, it has been pointed out by B. D. Nichols, co-author of SOLA algorithm, in a private communication that no multiple solutions have been found.

In conclusion, the model is able to predict both the transient flow and steady flow phenomena. The numerical stability problem can be solved by trial-and-error method with help of stability theorem. However, it should be noted that this is only a simpilfied model. Modifications may be necessary to fit specific problems.

VI RECOMMENDATION

To have a better simulation of liquid rotational molding, further studies are required to eliminate some of the assumptions and restrictions.

Recently a SOLA-VOF program (32) has been developed to include surface tension forces. It also has a flexible mesh generator to provide cells with various different sizes. This will alleviate the aspect ratio problem. The whole circumference can be approximated by the diagonals of the cells.

Constantly changing viscosity is a phenomenon associated with curing. As viscosity increases, the parameters and time increment should be changed. An automatic time-step control in SOLA-VOF to adjust the time increment is not enough. A parameters adjustment has to be added. The relationship of time-temperature-viscosity (12) had better be a separate program. Even though the MAC method is capable to handle heat transfer differential equations, it is believed to be over-entangled., to couple heat transfer, reaction heat and fluid mechanics The main reason is, again, numerical stability.

APPENDIX I

FINITE DIFFERENCE REPRESENTATION OF NAVIER-STOKES EQUATIONS

The approximation of Eqs. 14 a, b and 15 by finite difference and partial donor cell methods is illustrated in this appendix.

The gravity constants g_x and g_y need no approximation. The velocity terms $\partial u/\partial x$ and $\partial v/\partial y$ are taking the form of backward differencing, i.e., Eqs. 19 c, d. The transient and pressure terms $\partial u/\partial t$, $\partial v/\partial t$, $\partial p/\partial x$ and $\partial p/\partial y$ are expressed by forward differencing similar to Eqs. 17 c, d, for example,

$$\partial u_{i,j}/\partial t = (u_{i,j} - u_{i,j})/\delta t$$

$$\partial p_{i,j}/\partial x = (p_{i+1,j} - p_{i,j})/\delta x$$

The viscous fluxes $\nu(\partial^2 u/\partial x^2 + \partial^2 u/\partial y^2)$ and $\nu(\partial^2 v/\partial x^2 + \partial^2 v/\partial y^2)$ including second derivatives are represented by the central differencing, i.e., Eqs. 20 c, d. The partial donor cell method is applied only to convective fluxes.

A. UUX

The first derivative is given by central differencing.

$$\partial uu/\partial x = UUX = u(u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j})/\delta x$$

$$= (uu_{i+\frac{1}{2},j} - uu_{i-\frac{1}{2},j})/\delta x \qquad \text{(I-1)}$$

For the first term $uu_{i+\frac{1}{2},j}$, the partial donor cell defines

$$u = \left[(1+\bar{\alpha})u_{i,j} + (1-\bar{\alpha})u_{i+1,j}\right]/2 \qquad \text{(I-2)}$$

where $\bar{\alpha}$ = weight of donor cell with value $1 \geq \bar{\alpha} \geq -1$

IF $\bar{\alpha} = \alpha(\text{sign } u_{i+\frac{1}{2},j})$ and $1 \geq \alpha \geq 0$, then, it becomes

$$uu_{i+\frac{1}{2},j} = \left[u_{i+\frac{1}{2},j}\, u_{i,j} + \alpha(\text{sign } u_{i+\frac{1}{2},j})\, u_{i+\frac{1}{2},j}\, u_{i,j} + u_{i+\frac{1}{2},j}\, u_{i+1,j} - \alpha(\text{sign } u_{i+\frac{1}{2},j})\, u_{i+\frac{1}{2},j}\, u_{i+1,j}\right]/2 \qquad \text{(I-3)}$$

Since

$$(\text{sign } u_{i+\frac{1}{2},j})\ u_{i+\frac{1}{2},j} = |u_{i+\frac{1}{2},j}|$$

and

$$u_{i+\frac{1}{2},j} = (u_{i,j} + u_{i+1,j})/2$$

Eq. I-3 is finalized as

$$uu_{i+\frac{1}{2},j} = \left[(u_{i,j} + u_{i+1,j})^2 + \alpha|u_{i,j} + u_{i+1,j}|(u_{i,j} - u_{i+1,j})\right]/4 \tag{I-4}$$

For the second term $uu_{i-\frac{1}{2},j}$, the partial donor cell representation of Eq. I-5 is developed through steps similar to Eqs. I-2, I-3 and I-4. By setting

$$u = \left[(1+\bar{\alpha})u_{i-1,j} + (1-\bar{\alpha})u_{i,j}\right]/2 \qquad (1 \geq \bar{\alpha} \geq -1)$$

and

$$\bar{\alpha} = \alpha(\text{sign } u_{i-\frac{1}{2},j}) \qquad (1 \geq \alpha \geq 0)$$

then

$$uu_{i-\frac{1}{2},j} = \left[u_{i-\frac{1}{2},j}\ u_{i-1,j} + \alpha(\text{sign } u_{i-\frac{1}{2},j})u_{i-\frac{1}{2},j}\ u_{i-1,j} + u_{i-\frac{1}{2},j}\ u_{i,j} - \alpha(\text{sign } u_{i-\frac{1}{2},j})u_{i-\frac{1}{2},j}\ u_{i,j}\right]/2$$

Since

$$(\text{sign } u_{i-\frac{1}{2},j})\ u_{i-\frac{1}{2},j} = |u_{i-\frac{1}{2},j}|$$

and

$$u_{i-\frac{1}{2},j} = (u_{i-1,j} + u_{i,j})/2$$

therefore

$$uu_{i-\frac{1}{2},j} = \left[(u_{i-1,j} + u_{i,j})^2 + \alpha|u_{i-1,j} + u_{i,j}| \times (u_{i-1,j} - u_{i,j})\right]/4 \tag{I-5}$$

Combining Eqs. I-1, I-4, and I-5 results

$$UUX = \left[(u_{i,j} + u_{i+1,j})^2 + \alpha|u_{i,j} + u_{i+1,j}|(u_{i,j} - u_{i+1,j}) - (u_{i-1,j} + u_{i,j})^2 - \alpha|u_{i-1,j} + u_{i,j}|(u_{i-1,j} - u_{i,j})\right]/4\delta x \tag{27a}$$

B. UVY

Following central differencing scheme, the first derivative is

$$\partial uv/\partial y = UVY = u(v_{i+\frac{1}{2},j} - v_{i+\frac{1}{2},j-1})/\delta y$$

$$= (uv_{i+\frac{1}{2},j} - uv_{i+\frac{1}{2},j-1})/\delta y \qquad \text{(I-6)}$$

It should be noted that the X index of v terms is not i, but rather i + ½. The reason can be visualized in Fig. 31. The terms of the first derivative have to be on the same line, either horizontally or vertically. In Fig. 31(a), the terms for $\partial uu/\partial x$ are on a horizontal line. To represent $\partial uv/\partial y$ every term is on a vertically line as shown in Fig. 31(b).

Applying partial donor cell method to the first term $uv_{i+\frac{1}{2},j}$ in Eq. I-6,

$$u = [(1+\bar{\alpha})\, u_{i,j} + (1-\bar{\alpha})\, u_{i,j+1}]/2 \qquad (1 \geq \bar{\alpha} \geq -1)$$

and setting

$$\bar{\alpha} = \alpha(\text{sign } v_{i+\frac{1}{2},j}) \qquad (1 \geq \alpha \geq 0)$$

it becomes

$$uv_{i+\frac{1}{2},j} = [u_{i,j}\, v_{i+\frac{1}{2},j} + \alpha(\text{sign } v_{i+\frac{1}{2},j})v_{i+\frac{1}{2},j}\, u_{i,j}$$
$$+ u_{i,j+1}\, v_{i+\frac{1}{2},j} - \alpha(\text{sign } v_{i+\frac{1}{2},j})v_{i+\frac{1}{2},j}\, u_{i,j+1}]/2 \qquad \text{(I-7)}$$

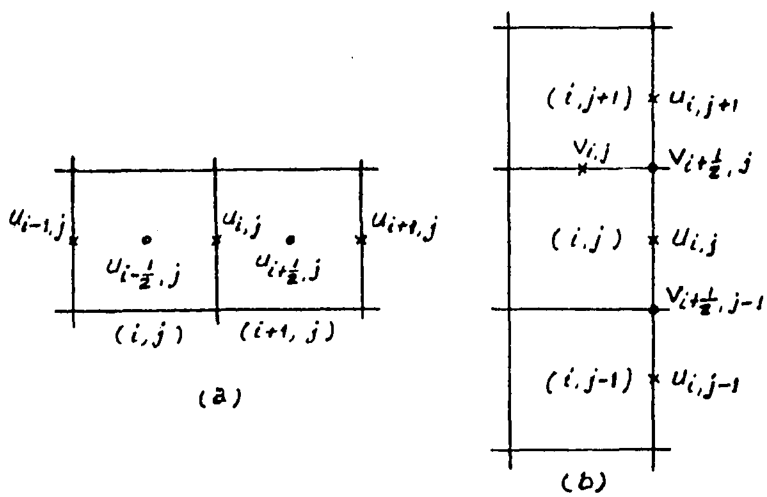


Fig. 31. Finite difference terms for (a) $\partial uu/\partial x$ and (b) $\partial uv/\partial y$.

By definition of

$$(\text{sign } v_{i+\frac{1}{2},j})\, v_{i+\frac{1}{2},j} = |v_{i+\frac{1}{2},j}|$$

and a centered convective flux

$$v_{i+\frac{1}{2},j} = (v_{i,j} + v_{i+1,j})/2,$$

Eq. I-7 results

$$uv_{i+\frac{1}{2},j} = [\,(u_{i,j} + u_{i,j+1})\,(v_{i,j} + v_{i+1,j}) + \alpha\,|v_{i,j} + v_{i+1,j}|\,(u_{i,j} - u_{i,j+1})]/4 \qquad \text{(I-8)}$$

Similarly, the second term $uv_{i+\frac{1}{2},j-1}$ of Eq. I-6 obtains the expression Eq. I-9. By setting

$$u = [\,(1+\bar{\alpha})u_{i,j-1} + (1-\bar{\alpha})\,u_{i,j}]/2 \qquad (1 \geq \bar{\alpha} \geq -1)$$

and

$$\bar{\alpha} = \alpha(\text{sign } v_{i+\frac{1}{2},j-1}) \qquad (1 \geq \alpha \geq 0)$$

and by definition

$$(\text{sign } v_{i+\frac{1}{2},j-1})\, v_{i+\frac{1}{2},j-1} = |\,v_{i+\frac{1}{2},j-1}\,|$$

then by setting

$$v_{i+\frac{1}{2},j-1} = (v_{i,j-1} + v_{i+1,j-1})/2$$

the result is

$$uv_{i+\frac{1}{2},j-1} = [(u_{i,j-1} + u_{i,j})\,(v_{i,j-1} + v_{i+1,j-1}) + \alpha|\,v_{i,j-1} + v_{i+1,j-1}\,|\,(u_{i,j-1} - u_{i,j})]/4 \qquad \text{(I-9)}$$

Combining Eqs. I-6, I-8 and I-9 yields

$$UVY = [(u_{i,j} + u_{i,j+1})\,(v_{i,j} + v_{i+1,j}) + \alpha|v_{i,j} + v_{i+1,j}|\,x\,(u_{i,j} - u_{i,j+1}) - (u_{i,j-1} + u_{i,j})\,(v_{i,j-1} + v_{i+1,j-1}) - \alpha|v_{i,j-1} + v_{i+1,j-1}|(u_{i,j-1} - u_{i,j})]/4\delta y \qquad \text{(27b)}$$

C. <u>UVX</u>

The derivation for $\partial uv/\partial x$ is basically the same step by step of operation as for $\partial uv/\partial y$. From Fig. 32(a), the terms on a horizontal line are to be used in the central differencing.

$$\partial uv/\partial x = UVX = v(u_{i,j+\frac{1}{2}} - u_{i-1,j+\frac{1}{2}})/\delta x$$

$$= (vu_{i,j+\frac{1}{2}} - vu_{i-1,j+\frac{1}{2}})/\delta x \qquad \text{(I-10)}$$

For the first term $vu_{i,j+\frac{1}{2}}$, setting

$$v = [(1+\bar{\alpha})v_{i,j} + (1-\bar{\alpha})v_{i+1,j}]/2 \qquad (1 \geq \bar{\alpha} \geq -1)$$

and

$$\bar{\alpha} = \alpha(\text{sign } u_{i,j+\frac{1}{2}}) \qquad (1 \geq \alpha \geq 0)$$

then

$$vu_{i,j+\frac{1}{2}} = [u_{i,j+\frac{1}{2}}(v_{i,j} + v_{i+1,j})$$
$$+ \alpha(\text{sign } u_{i,j+\frac{1}{2}})u_{i,j+\frac{1}{2}}(v_{i,j} - v_{i+1,j})]/2 \qquad \text{(I-11)}$$

By definition

$$(\text{sign } u_{i,j+\frac{1}{2}})u_{i,j+\frac{1}{2}} = |u_{i,j+\frac{1}{2}}|$$

then, setting

$$u_{i,j+\frac{1}{2}} = (u_{i,j} + u_{i,j+1})/2$$

Eq. I-11 becomes

$$vu_{i,j+\frac{1}{2}} = [(u_{i,j} + u_{i,j+1})(v_{i,j} + v_{i+1,j})$$
$$+ \alpha|u_{i,j} + u_{i,j+1}|(v_{i,j} - v_{i+1,j})]/4 \qquad \text{(I-12)}$$

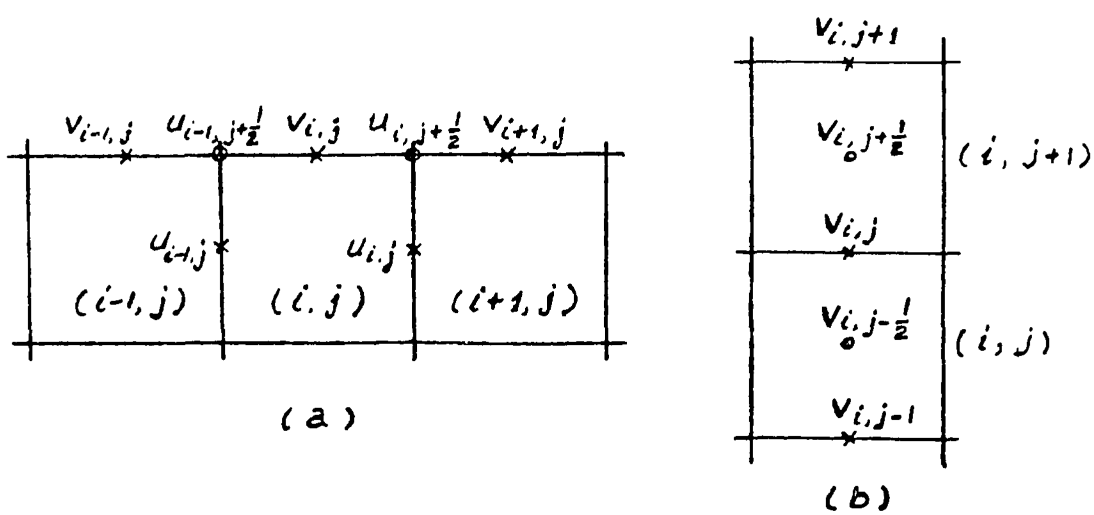


Fig. 32. Finite difference terms for (a) $\partial uv/\partial x$ and (b) $\partial vv/\partial y$.

For the second term $vu_{i-1,j+\frac{1}{2}}$ in Eq. I-10, the similar procedures are followed to yield Eq. I-13.

By setting

$$v = [(1+\bar{\alpha})v_{i-1,j} + (1-\bar{\alpha})v_{i,j}]/2 \qquad (1 \geq \bar{\alpha} \geq -1)$$

and

$$\bar{\alpha} = \alpha\,(\text{sign } u_{i+1,j+\frac{1}{2}}) \qquad (1 \geq \alpha \geq 0)$$

then

$$(\text{sign } u_{i-1,j+\frac{1}{2}})\, u_{i-1,j+\frac{1}{2}} = |u_{i-1,j+\frac{1}{2}}|$$

setting

$$u_{i-1,j+\frac{1}{2}} = (u_{i-1,j} + u_{i,j+1})/2$$

then

$$vu_{i-1,j+\frac{1}{2}} = [(u_{i-1,j} + u_{i-1,j+1})\,(v_{i-1,j} + v_{i,j}) + \alpha|u_{i-1,j} + u_{i-1,j+1}|\,(v_{i-1,j} - v_{i,j})]/4 \qquad \text{(I-13)}$$

Combining Eqs. I-10, I-12 and I-13 obtains

$$\begin{aligned} UVX = [&(u_{i,j} + u_{i+1,j})\,(v_{i,j} + v_{i+1,j}) \\ &+\alpha|u_{i,j} + u_{i,j+1}|(v_{i,j} - v_{i-1,j}) \\ &-(u_{i-1,j} + u_{i-1,j+1})\,(v_{i-1,j} + v_{i,j}) \\ &-\alpha|u_{i-1,j} + u_{i-1,j+1}|\,(v_{i-1,j} - v_{i,j})]/4\delta x \end{aligned} \qquad \text{(27d)}$$

D. <u>VVY</u>

This derivation is closely similar to that in section A for $\partial uu/\partial x$. All the terms involved are shown in Fig. 32(b).

$$\begin{aligned} \partial vv/\partial y = VVY &= v(v_{i,j+\frac{1}{2}} - v_{i,j-\frac{1}{2}})/\delta y \\ &= (vv_{i,j+\frac{1}{2}} - vv_{i,j-\frac{1}{2}})/\delta y \end{aligned} \qquad \text{(I-14)}$$

For the first term $vv_{i,j+\frac{1}{2}}$, since

$$v = [(1+\bar{\alpha})v_{i,j} + (1-\bar{\alpha})v_{i,j+1}]/2 \qquad (1 \geq \bar{\alpha} \geq -1)$$

and

$$\bar{\alpha} = \alpha\,(\text{sign } v_{i,j+\frac{1}{2}}) \qquad (1 \geq \alpha \geq 0)$$

and by definition

$$(\text{sign } v_{i,j+\frac{1}{2}})v_{i,j+\frac{1}{2}} = |v_{i,j+\frac{1}{2}}|$$

and by centered differencing

$$v_{i,j+\frac{1}{2}} = (v_{i,j} + v_{i,j+1})/2$$

the result is

$$vv_{i,j+\frac{1}{2}} = [(v_{i,j} + v_{i,j+1})^2 + \alpha|v_{i,j} + v_{i,j+1}| \times (v_{i,j} - v_{i,j+1})]/4 \tag{I-15}$$

Similarly, the second term $vv_{i,j-\frac{1}{2}}$ is expanded and consolidated. Since

$$v = [(1+\bar{\alpha})v_{i,j-1} + (1-\bar{\alpha})v_{i,j}]/2 \qquad (1 \geq \bar{\alpha} \geq -1)$$

and

$$\bar{\alpha} = \alpha\ (\text{sign } v_{i,j-\frac{1}{2}}) \qquad (1 \geq \alpha \geq 0)$$

and by definition

$$(\text{sign } v_{i,j-\frac{1}{2}})v_{i,j-\frac{1}{2}} = |v_{i,j-\frac{1}{2}}|$$

and by centered differencing

$$v_{i,j-\frac{1}{2}} = (v_{i,j-1} + v_{i,j})/2$$

the result is

$$vv_{i,j-\frac{1}{2}} = [(v_{i,j-1} + v_{i,j})^2 + \alpha|v_{i,j-1} + v_{i,j}| \times (v_{i,j-1} - v_{i,j})]/4 \tag{I-16}$$

Combining Eqs. I-14, I-15 and I-16 results

$$\begin{aligned} VVY = [&(v_{i,j} + v_{i,j+1})^2 + \alpha|v_{i,j} + v_{i,j+1}| \times \\ &(v_{i,j} - v_{i,j+1}) - (v_{i,j-1} + v_{i,j})^2 \\ &- \alpha|v_{i,j-1} + v_{i,j}|\ (v_{i,j-1} - v_{i,j})]/4\delta y \end{aligned} \tag{27e}$$

APPENDIX II

SURFACE LOCATION AND SLOPE

In this appendix, several cases of surface determination are discussed. Also described is the slope of the surface. Since one quadrant is mirror image of any other, the calculation is performed for third quadrant only.

A. Surface Location

In Fig. 33(a), a curved boundary line is crossing a cell (i, j), which is located by BX and BY.

$$BX = (i - 1)\delta x \qquad \text{(II - 1)}$$

From geometrical relationship, it yields

$$(AR - BY)^2 = AR^2 - (AR - BX)^2$$

Rearranging the equation for BY, it results

$$BY = AR - [\,2(AR)(BX) - BX^2]^{1/2} \qquad \text{(II - 2)}$$

At least four different relationships between JBL and JBR, which are defined in the followings, should be discussed as shown in Fig. 33(b).

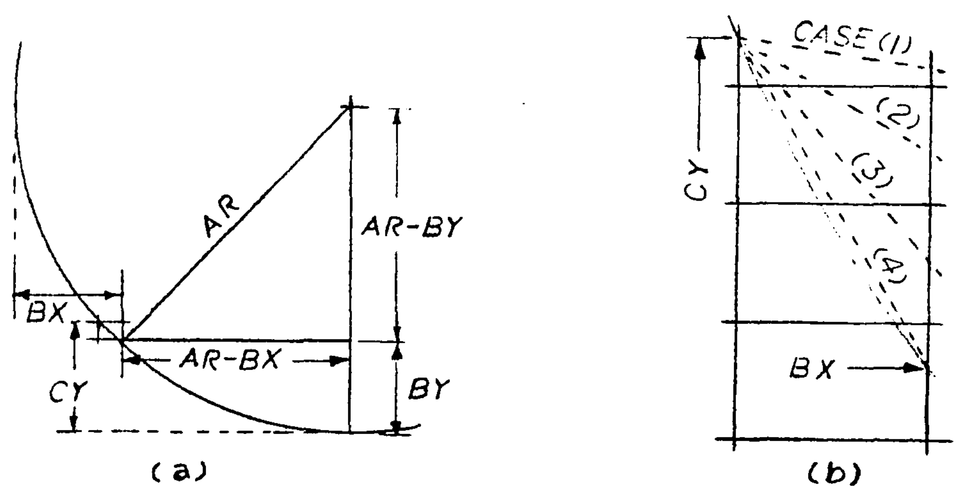


Fig. 33 Computation of surface location.

$BL = CY/\delta y$

$JBL = INT\ (BL)$

$BR = BY/\delta y$

$JBR = INT\ (BR)$

1. JBL = JBR (Fig. 34(a))

   If $(BR - JBR) + (BL - JBL) \geq 1.5$, JBR +1 is the boundary cell.

   Otherwise, JBL + 2 is the boundary.

2. JBL - 1 = JBR (Fig. 34(b))

   $(BL - JBL)/\ [1-(BR - JBR)] = (1-PX)/PX = BK$

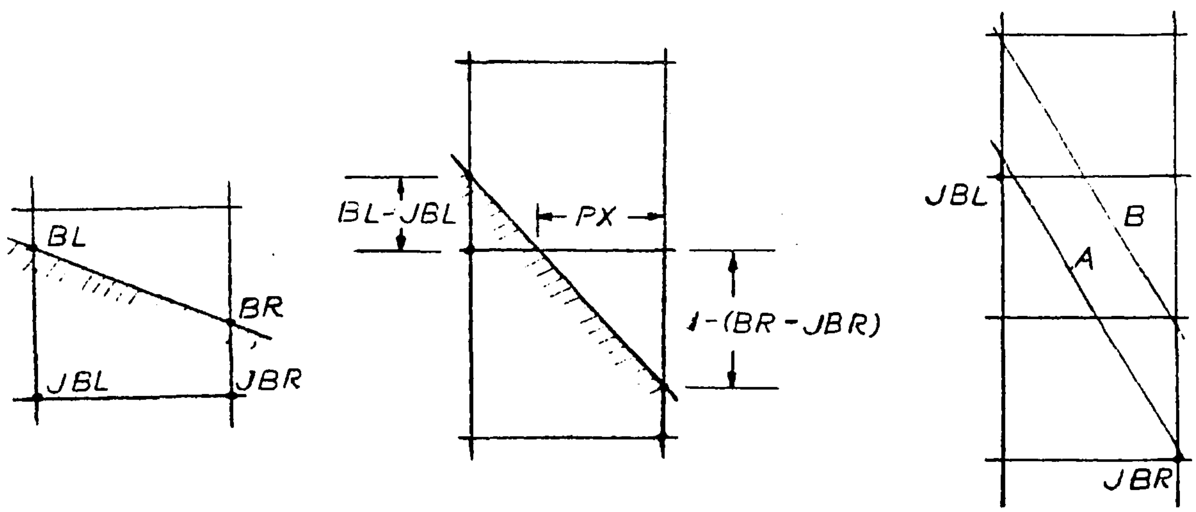
   Let $RBX = 1/(1 + BK) = PX$

   If $0.5[1 - (BR - JBR)]/(1 + BK) \geq \frac{1}{4}$, JBR + 1 is the boundary.

   Otherwise, JBL + 1 is the boundary

3. JBL - 2 = JBR

   In either line A or line B of Fig. 34(c) the boundary is the cell of JBL + 2. The triangular area in the cell of JBL + 1 is always less than ¼ of a cell.



(a) (b) (c)

Fig. 34 Determination of the boundary.
(a) JBL = JBR, (b) JBL - 1 = JBL, (c) JBL - 2 = JBR.

4. JBL - 3 $\geq$ JBR

For a steep slope boundary the boundary cell is at the midpoint of BY and CY.

B. <u>Slope of Boundary</u>

By connecting the center of drum, 0, to mid-point of bottom surface of cell i and defining X_i = BX - X/2, the angle θ can be computed.

$$\theta = \tan^{-1}\left[(AR - X_i)/(AR - HB)\right]$$

The slope of boundary at cell i is

$$SLOB_i = \tan\theta$$

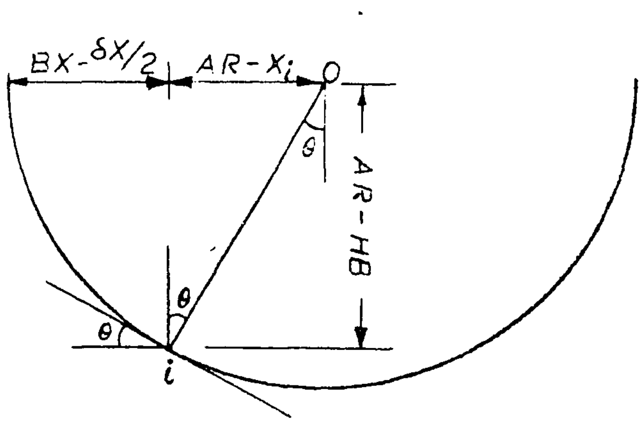


Fig. 35 Slope of boundary.

APPENDIX III

ADJUSTMENTS OF CELL PRESSURE AND VELOCITY

The derivations of Eqs. 29 and 30a - d in Sec. IV.C Velocity Iteration are described in this appendix.

The momentum equation 25a is re-written in two groups of variables; one group contains pressure item, another group does not.

$$u^{o}_{i,j} = u_{i,j} + \delta t(g_x - UUX - UVY - VISU) - (p_{i+1,j} - p_{i,j})\,\delta t/\delta x$$

$$= UCON_{i,j} - (p_{i+1,j} - p_{i,j})\delta t/\delta x \qquad \text{(III-1a)}$$

where $UCON_{i,j} = u_{i,j} + \delta t(g_x - UUX - UVY - VISU)$

Since $UCON_{i,j}$ consists of the terms in old time level, it is a constant in the new time level. Similary, Eq. 25b ia re-written.

$$v^{o}_{i,j} = VCON_{i,j} - (p_{i,j+1} - p_{i,j})\delta t/\delta y \qquad \text{(III-1b)}$$

Two more equations in the same form are needed.

$$u^{o}_{i-1,j} = UCON_{i-1,j} - (p_{i,j} - p_{i-1,j})\delta t/\delta x \qquad \text{(III-1c)}$$

$$v^{o}_{i,j-1} = VCON_{i,j-1} - (p_{i,j} - p_{i,j-1})\delta t/\delta x \qquad \text{(III-1d)}$$

From the above four equations III-1a thru 1d, the following derivatives are resulted:

$$\partial u^{o}_{i,j} \;/\partial p_{i,j} = \delta t/\delta x \qquad \text{(III-2a)}$$

$$\partial u^{o}_{i-1,j}/\partial p_{i,j} = -\delta t/\delta x \qquad \text{(III-2b)}$$

$$\partial v^{o}_{i,j} \;/\partial p_{i,j} = \delta t/\delta y \qquad \text{(III-2c)}$$

$$\partial v^{o}_{i-1,j}/\partial p_{i,j} = -\delta t/\delta y \qquad \text{(III-2d)}$$

Eqs. III-2a thru 2d can be expressed in difference form in respect to iteration cycle.

$$u^{n+1}_{i,j} = u^{n}_{i,j} + (p^{n+1}_{i,j} - p^{n}_{i,j})\delta t/\delta x$$

$$= u^{n}_{i,j} + \delta p\;\delta t/\delta x \qquad \text{(III-3a)}$$

$$u^{n+1}_{i-1,j} = u^{n}_{i-1,j} - (p^{n+1}_{i,j} - p^{n}_{i,j})\delta t/\delta x$$
$$= u^{n}_{i-1,j} - \delta p \; \delta t/\delta x \qquad \text{(III-3b)}$$

$$v^{n+1}_{i,j} = v^{n}_{i,j} + (p^{n+1}_{i,j} - p^{n}_{i,j})\delta t/\delta y$$
$$= v^{n}_{i,j} + \delta p \; \delta t/\delta y \qquad \text{(III-3c)}$$

$$v^{n+1}_{i,j-1} = v^{n}_{i,j-1} - (p^{n+1}_{i,j} - p^{n}_{i,j})\delta t/\delta y$$
$$= v^{n}_{i,j-1} - \delta p \; \delta t/\delta y \qquad \text{(III-3d)}$$

where $\delta p = p^{n+1}_{i,j} - p^{n}_{i,j}$

n, $n+1$ = Iteration cycles in new time level.

The velocities u and v are both in the new time level but different iteration cycles as indicated by the superscripts. Substituting Eqs. III - 3a thru 3d into discrepancy equation similar to Eq. 28 yields

$$D^{n+1} = (u^{n+1}_{i,j} - u^{n}_{i-1,j})/\delta x + 2\,\delta p\;\delta t/\delta x^2$$
$$+ (v^{n}_{i,j} - v^{n}_{i,j-1})/\delta y + 2\,\delta p\;\delta t/\delta y^2 \qquad \text{(III-4)}$$

Because a convergence has not been reached in the n th iteration, the discrepancy is

$$D^{n} = (u^{n}_{i,j} - u^{n}_{i-1,j})/\delta x + (v^{n}_{i,j} - v^{n}_{i,j-1})/\delta y$$

Eq. III-4 is simplified.

$$D^{n+1} = D^{n} + \delta p(2\delta t/\delta x^2 + 2\delta t/\delta y^2) \qquad \text{(III-5)}$$

To diminish the discrepancy in (n+1)th iteration, i.e., $D^{n+1} = 0$, the pressure is adjusted by an amount of δp.

$$\delta p = - D^{n}/(2\delta t/\delta x^2 + 2\delta t/\delta y^2) \qquad \text{(III-6)}$$

$$p^{n+1} = p^{n} + \delta p \qquad \text{(III-7)}$$

Eq. III-6 is identical to Eq. 29. Eqs. III-3 a, b, c, d are the same as Eqs. 30 a - d, respectively.

APPENDIX IV

PRESSURE ADJUSTMENT AT BOUNDARY

This appendix details the pressure adjustment at exterior of the mesh, either rigid boundary or free surface, i.e., Eqs. 30a - d.

A   <u>Free Surface</u>

The pressure at free surface cell is computed from the cells above and below the surface by a linear interpolation.

A.1   <u>Top Free Surface</u>

The pressure at mid-surface is p_s .

$$p_s = p_{i,jT} - (p_{i,jT-1} - p_{i,jT})\,\Delta y/\delta y$$

$$= (1+\Delta y/\delta y)p_{i,jT} - p_{i,jT-1}\,\Delta y/\delta y$$

$$= p_{i,jT}/\eta_T + (1 - 1/\eta_T)p_{i,jT-1} \qquad \text{(IV-1)}$$

where $1/\eta_T = 1 + \Delta y/\delta y$

Re-arranging Eq. IV-1 for $p_{i,jT}$ results

$$p_{i,jT} = \eta_T\, p_s + (1 - \eta_T)p_{i,jT-1} \qquad \text{(IV-2)}$$

The factor of surface location is included in the η_T term.

$$\eta_T = \delta y/(\delta y + \Delta y)$$

$$= \delta y/(\delta y + H_i - H_c)$$

$$= \delta y/[\delta y + H_i - (jT - 1.5)\,\delta y]$$

where $\Delta y = H_i - H_c$

$H_c = (jT - 1.5)\delta y$

When uniform pressure is applied at the free surface, it is convenient to set $p_s = 0$. Then, Eq. IV-2 becomes

$$p_{i,jT} = (1 - \eta_T)p_{i,jT-1} \qquad \text{(IV-3)}$$

A value of $p_{i,jT}$ computed by Eq. IV-3 is the correct one. Therefore, the pressure adjustment, δp, is

$$\delta p = (1 - \eta_T)p_{i,jT-1} - p_{i,jT} \qquad \text{(31a)}$$

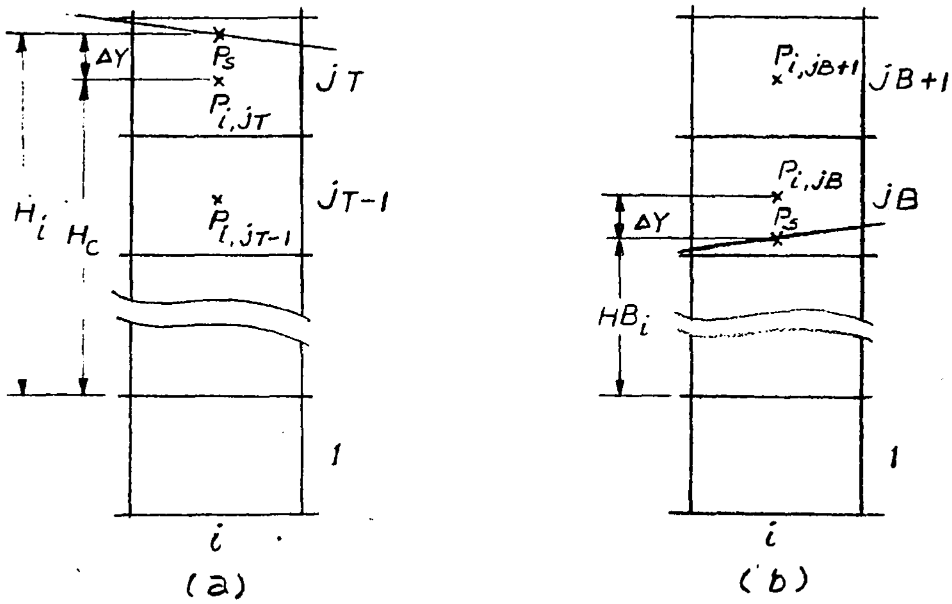


Fig. 36 Pressure adjustment at free surface

(a) Top free surface, (b) Bottom free surface

A.2 Bottom Free Surface

The steps of dirivation is similar to top free surface. The terms are shown in Fig. 36(b).

$$p_s = p_{i,jB} + (p_{i,jB} - p_{i,jB+1})\,\Delta y/\delta y$$
$$= (1 + \Delta y/\delta y)p_{i,jB} - p_{i,jB+1}\,\Delta y/\delta y$$
$$= p_{i,jB}/\eta_B + (1 - 1/\eta_B)p_{i,jB+1} \quad \text{(IV-4)}$$

where $1/\eta_B = 1 + \Delta y/\delta y$

and $\eta_B = \delta y/(\Delta y + \delta y)$

$$= \delta y/[\delta y + (jB - 1.5)\delta y - HB_i]$$

Re-arranging Eq. IV-4 for $p_{i,jB}$,

$$p_{i,jB} = \eta_B p_s + (1 - \eta_B)p_{i,jB+1} \quad \text{(IV-5)}$$

When $p_s = 0$, A correct $p_{i,jB}$ is

$$p_{i,jB} = (1 - \eta_B)p_{i,jB+1}$$

The pressure adjustment is

$$\delta p = (1 - \eta_B)p_{i,jB+1} - p_{i,jB} \quad \text{(31b)}$$

B. Rigid Surface

The surface pressure at rigid surface will make the normal velocity equal to zero.

$$u_n = u \cdot n = 0$$

$$= (iu - jv)\ (\ -i \sin\theta + j \cos\theta)$$

$$= -u \sin\theta + v \cos\theta$$

$$u_n/\cos\theta = -\ u \tan\theta + v \qquad \text{(IV-6)}$$

B.1 Bottom Rigid Surface

The horizontal velocity at mid-surface is the average of two cell velocities.

$$u = (u_{i,jB} + u_{i-1,jB})/2 \qquad \text{(IV-7)}$$

The vertical velocity can be found by interpolation shown in Fig. 37.

$$v = v_{i,jB-1} + (v_{i,jB} - v_{i,jB-1})\Delta y/\delta y \qquad \text{(IV-8)}$$

The slope of surface is

$$\tan\theta = (HB_{i+1} - HB_{i-1})/2\delta x \qquad \text{(IV-9)}$$

Substituting Eqs. IV-7, IV-8 and IV-9 into Eq. IV-6 results

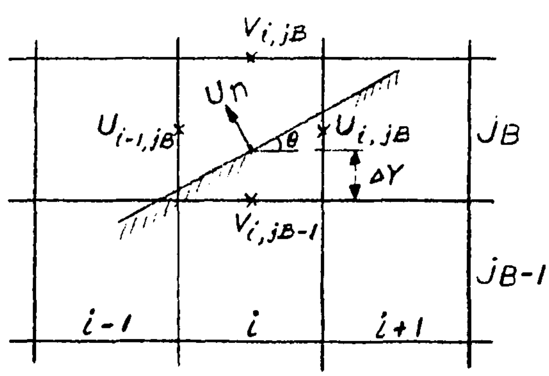


Fig. 37 Pressure adjustment at bottom rigid surface.

$$u_n/\cos\theta = -(u_{i,jB} + u_{i-1,jB})(HB_{i+1} - HB_{i-1})/4\delta x$$
$$+ v_{i,jB}\,\Delta y/\delta y + v_{i,jB-1}(1 - \Delta y/\delta y) \qquad \text{(IV-10)}$$

Since $v_{i,jB-1}$ is inside the rigid boundary, it would be eliminated by equation of continuity, i.e.,

$$v_{i,jB-1} = v_{i,jB} + (u_{i,jB} - u_{i-1,jB})\delta y/\delta x \qquad \text{(IV-11)}$$

Eq. IV-10 is expanded and re-arranged to the following:

$$u_n/\cos\theta = -(u_{i,jB} + u_{i-1,jB})(HB_{i+1} - HB_{i-1})/4\delta x + v_{i,jB}$$
$$+ (u_{i,jB} - u_{i-1,jB})(1 - \Delta y/\delta y)(\delta y/\delta x) \qquad \text{(IV-12)}$$

The only term to be computed in the right hand size of Eq. IV-12 is y.

$$1 - \Delta y/\delta y = 1 - [HB_i - (jB-2)\delta y]/\delta y = (jB-1) - HB_i/\delta y$$
$$= \lambda_B \qquad \text{(IV-13)}$$

The velocity is function of pressure as expressed in Eq. III-2a. The increment of pressure can be found by Newton-Ralphson method.

$$\delta p = -u_n/(\partial u_n/\partial p)$$
$$= -(u_n/\cos\theta)/[\partial(u_n/\cos\theta)/\partial p] \qquad \text{(IV-14)}$$

The denominator is, from Eqs. IV-12 and IV-13,

$$\partial(u_n/\cos\theta)/\partial p = -(\partial u_{i,jB}/\partial p + \partial u_{i-1,jB}/\partial p)(HB_{i+1} - HB_{i-1})/4\delta x$$
$$+ \partial v_{i,jB}/\partial p + (\partial u_{i,jB}/\partial p - \partial u_{i-1,jB}/\partial p)\lambda_B(\delta y/\delta x) \qquad \text{(IV-15)}$$

where

$$p = p_{i,jB}$$

Substituing Eqs. III-2a thru d into Eq. IV-15 yields

$$\partial(u_n/\cos\theta)/\partial p = \delta t/\delta y + 2\lambda_B(\delta t/\delta x)(\delta y/\delta x)$$
$$= [1 + 2\lambda_B(\delta y/\delta x)^2]\delta t/\delta y \qquad \text{(IV-16)}$$

Substituting Eqs. IV-12, IV-13 and IV-16 into IV-14 results Eq. 31d.

B.2 Top Rigid Surface

Similar to the step of derivation in above section for bottom rigid surface, the terms are shown in Fig. 38.

The velocities at mid-surface are

$$u = (u_{i,jT} + u_{i-1,jT})/2 \tag{IV-17}$$

$$v = v_{i,jT-1} + (v_{i,jT} - v_{i,jT-1})\Delta y/\delta y$$

$$= v_{i,jT-1}(1 - \Delta y/\delta y) + v_{i,jT}\Delta y/\delta y \tag{IV-18}$$

The slope of surface is

$$\tan\theta = (H_{i+1} - H_{i-1})/2\delta x \tag{IV-19}$$

The term of $v_{i,jT}$ is eliminated by continuity equation.

$$v_{i,jT} = v_{i,jT-1} - (u_{i,jT} - u_{i-1,jT})(\delta y/\delta x) \tag{IV-20}$$

Eq. IV-18 becomes

$$v = v_{i,jT-1} - (u_{i,jT} - u_{i-1,jT})(\delta y/\delta x)(\Delta y/\delta y) \tag{IV-21}$$

$\Delta y/\delta y$ is the only term requiring computation in right hand side of Eq. IV-21.

$$\Delta y/\delta y = [H_i - (jT-2)\delta y]/\delta y = H_i/\delta y - (jT-2)$$

$$= \lambda_T \tag{IV-22}$$

Eq. IV-6 becomes

$$u_n/\cos\theta = -(u_{i,jT} + u_{i-1,jT})(H_{i+1} - H_{i-1})/4\delta x$$

$$+ v_{i,jT-1} - \lambda_T (u_{i,jT} - u_{i-1,jT})\delta y/\delta x \tag{IV-23}$$

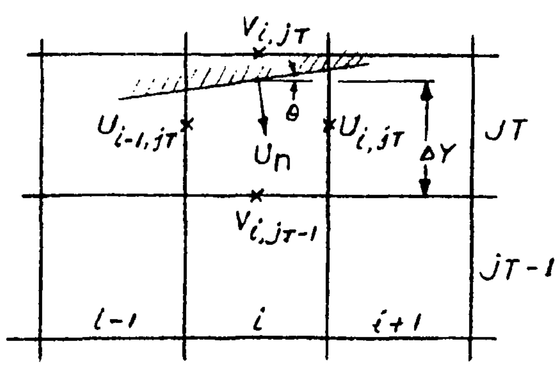
The denominator of Eq. IV-14 is

$$\partial(u_n/\cos\theta)/\partial p = -\delta t/\delta y - 2\lambda_T(\delta t/\delta x)(\delta y/\delta x)$$

$$= -[1 + 2\lambda_T(\delta y/\delta x)^2]\delta t/\delta y \tag{IV-24}$$

Substituting Eqs. IV-23 and IV-24 into Eq. IV-14 results in Eq. 31 c.

Fig. 38 Pressure adjustment at top rigid surface



APPENDIX V

FREE SURFACE LOCATION

To Express Eq. 32 in finite difference form for top free surface and bottom free surface is explained in this appendix.

A. Top Free Surface

The vertical velocity at free surface is found by interpolation shown in Fig. 39

$$v = v_{i,jT-1} + (v_{i,jT} - v_{i,jT-1})\Delta y/\delta y$$

$$= v_{i,jT}\,\Delta y/\ y + v_{i,jT-1}\,(1 - \Delta y/\delta y)$$

$$= v_{i,jT}\,\lambda_T + v_{i,jT-1}\,(1 - \lambda_T) \qquad \text{(V-1)}$$

where λ_T = Eq. IV-22

The horizontal velocity is the average of upstream and downstream ones.

$$u = (u_{i,jT} + u_{i-1,jT})/2 \qquad \text{(V-2)}$$

Similar to Eq. I-2, the H terms are defined by partial donor cell method.

$$H = H_{i+1/2} - H_{i-1/2} \qquad \text{(V-3)}$$

$$H_{i+1/2} = [(1+\bar{\gamma})\,H_i + (1-\bar{\gamma})\,H_{i+1}]/2 \qquad \text{(V-4)}$$

$$H_{i-1/2} = [(1+\bar{\gamma})\,H_{i-1} + (1-\bar{\gamma})\,H_i]/2 \qquad \text{(V-5)}$$

where $\bar{\gamma}$ = weight of donor cell with value $1 \geq \bar{\gamma} \geq -1$

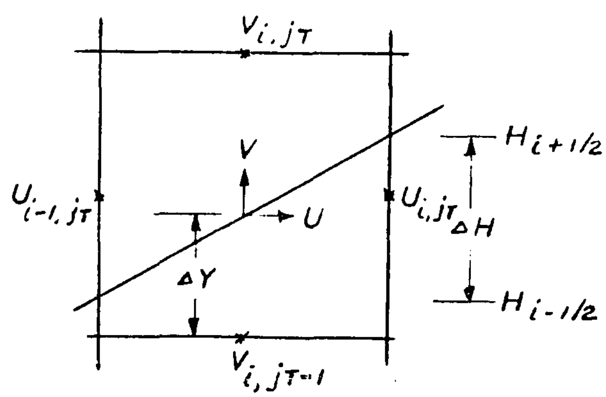


Fig. 39 Location of top free surface

$$u\Delta H = (u_{i,jT} + u_{i-1,jT})\,(H_{i+1/2} - H_{i-1/2})/2$$

$$= (u_{i,jT} + u_{i-1,jT})\,[H_{i+1} - H_{i-1} - \bar{\gamma}(H_{i+1} - 2H_i + H_{i-1})]/4 \qquad \text{(V-6)}$$

Same reasoning as in Appendix I, if

$$\bar{\gamma} = \gamma\,(\text{sign } u_{i-1/2,jT}) \qquad (1 \geq \gamma \geq 0)$$

Eq. V-6 is simplified.

$$u\Delta H = [(u_{i,jT} + u_{i-1,jT})\,(H_{i+1} - H_{i-1}) - \gamma\,|u_{i,jT} + u_{i-1,jT}|\,x\,(H_{i+1} - 2H_i + H_{i-1})]/4 \qquad \text{(V-7)}$$

Substituting Eqs. V-1 and V-7 into Eq. 32 results Eq. 33 a

B. Bottom Free Surface

The procedure of derivation of Eq. 33b is similar to the above section. The vertical velocity is

$$v = v_{i,jB-1} + (v_{i,jB} - v_{i,jB-1})\Delta y/\delta y$$

$$= v_{i,jB}\,\Delta y/\delta y + v_{i,jB-1}\,(1 - \Delta y/\delta y)$$

$$= (1 - \lambda_B)\,v_{i,jB} + \lambda_B\,v_{i,jB-1} \qquad \text{(V-8)}$$

where λ_B = Eq. IV-13

The horizontal velocity is

$$u = (u_{i,jB} + u_{i-1,jB})/2 \qquad \text{(V-9)}$$

The partial donor cell scheme is applied to HB terms, as shown in Fig. 40.

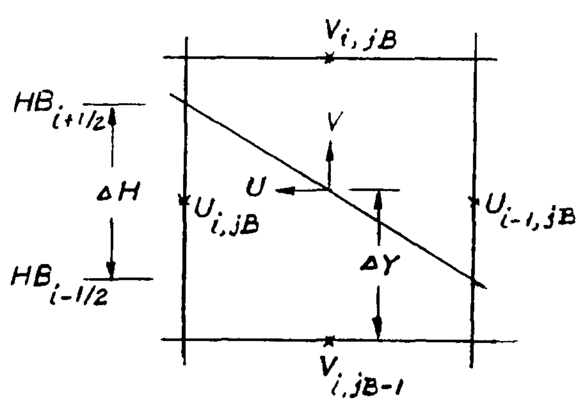


Fig. 40 Location of bottom free surface.

$$\Delta H = HB_{i+1/2} - HB_{i-1/2} \tag{V-10}$$

$$HB_{i+1/2} = [(1 + \bar{\gamma})\ HB_i + (1 - \bar{\gamma})\ HB_{i+1}]/2 \tag{V-11}$$

$$HB_{i-1/2} = [(1 + \bar{\gamma})\ HB_{i-1} + (1 - \bar{\gamma})\ HB_i]/2 \tag{V-12}$$

Substituting Eqs. V-11 and V-12 into Eq. V-10 and setting

$$\bar{\gamma} = \gamma\ (\text{sign}\ u_{i-1/2,jB})$$

result the following expression:

$$u\Delta H = [(u_{i,jB} + u_{i-1,jB})\ (HB_{i+1} - HB_{i-1}) - \gamma\,|u_{i,jB} + u_{i-1,jB}|\ (HB_{i+1} - 2HB_i + HB_{i-1})]/4 \tag{V-13}$$

Again, substituting Eqs. V-8 and V-13 into Eq. 32 yields Eq. 33b.

APPENDIX VI

COMPARISON OF DIMENSION APPROXIMATION IN EXAMPLE

Because of an approximation of physical dimension taken in the example, its effect has to be checked and be sure in negligible range.

A. Excluded Area

The excluded area can be found from table of circular segments (33) as shown in Fig. 41.

$R = 6.4$ cm

$h = (5 - 2)\delta x = 0.6$ cm

$h/R = 0.09375$

From the table of circular segments, the area is found

shaded area/$R^2 = 0.0533$

Because there are two shaded areas,

ratio of excluded area $= 2(0.0533)R^2/\pi R^2 = 3.4\%$

The excluded area is 3.4% of the cylinder.

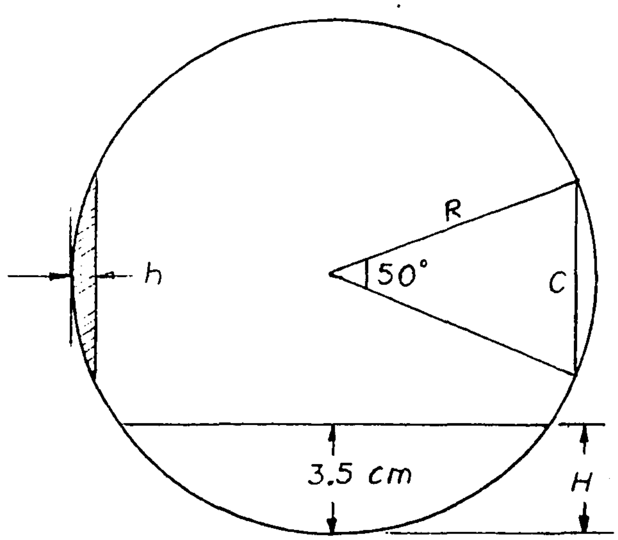


Fig. 41 Physical dimension of example problem.

B. Shortening of Circumference

The circumference of flat wall is also calculated from table of circular segments (33). As ratio of h/R is 0.09375 in above section, it is found

$c/R = 0.845$

Therefore, the flat wall (chord) is 5.41 cm and the central angle is 50°.

The segment of circumference is

$2\pi R\ (50/360) = 5.58$

The difference between the complete and the shortened circumferences is

$2(5.58 - 5.41)/2\pi R = 0.0085$

or 0.85%, which is negligible.

C. Fluid Occupied Area

The segment area occupied by fluid is computed by the same table (33) as in the above section A.

$H/R = 3.5/6.4 = 0.547$

$\text{area}/R^2 = 0.695$

$\text{area} = 0.695(6.4)^2 = 28.47\ \text{cm}^2$

The result of numerical approximation is 28.32 cm^2, or 99.5% of the theoretical value.

D. Film Thickness

The film thickness, b, is computed by assuming uniform thickness around the circumference.

$b = 28.32/2\pi R = 0.7\ \text{cm}$

The ratio of thickness to the radius of drum is

$b/R = 0.11$

Which is smaller than 20% but enough to cover more than a layer of single cell, i,e., 0.4 cm thickness in this example.

E. <u>Time To Cover The Whole Circumference</u>

In the example, the fluid travels from I = 59 to the upper half and returns to the lower half. Two portions in the lower half meet at I = 31. The total distance can be divided into five parts, as listed in the following table.

Table III

Total Traveling in the Example

Parts	From I =	To I =	Distance (cm)
(1)	59	62	*1
(2)	62	70	5.41 *2
(3)	70	127	14.51 *3
(4)	127	5	5.41 *2
(5)	5	31	7.25 *1

Notes:

1. The total number of cells traveled thru is 29, which is approximated half of the distance in part (3)
2. The chord length computed in sec. (b)
3. The segment for central angle 130^o is $2\pi R(130/360) = 14.51$ cm

The total traveling is approximately 32.58 cm. The minimum time to cover the whole circumference is 32.58/220 = 0.148 sec for rotating velocity 220 cm/sec. That is 0.148/0.0005 = 296 cycle. The numerical method results a complete ring at 327th cycle.

APPENDIX VII

NUMERICAL METHODS FOR FLUID DYNAMICS

As the physical law in fluid dynamics is expressed by differential equations, the best method of solution is including the most of theoretical method and the least of premanipulation, such as complex variables. But when the problem is more complex, more approximate procedure, e.g., boundary layer, perturbation method, etc., becomes necessary. To even more complicate problem involving odd boundary shape or three dimensions, only the direct numerical method is applicable. In this very last catagory of problems, a question often raised is how to select a numerical method. This appendix is discussing the criteria of numerical modeling and the applications of the important numerical approaches.

Harlow (47) compiled and classified the numerical methods for transient flow according to the flow speed, coordinate systems, and solution methods. For the high-speed flow, several approaches including Eulerian, Lagrangian, combined Eulerian and Lagrangian, Fourier series, Monte Carlo method, etc., have been tried. For the low-speed flows, the Eulerian approach is dominant. Other methods, such as Lagrangian and Fourier solution, need more studies to improve and extend applications.

The Eulerian approach allows the fluid undergoing great distortions without loss of accuracy, but it looses the sharp definition of contact surface. On the contrary, Lagrangian method can easily apply fine zoning at the interface and arbitrary curve shape of rigid boundary, but it is less accurate when the cells are hgihly distorted. In the Eulerian approach, the primitive variable method has the advantage of easy applying free surface boundary condition. This gain is offset by the difficulty of accurate pressure iteration. The vorticity-stream function method eliminates the problems of pressure term and continuity equation. However, it is difficult to encounter with unknown boundary condition on vorticity.

A series of comparisons among nine methods for incompressible, viscous, steady flow in a driven cavity was prepared by Langley Research Center (48). The Reynolds number of the test case was 100. The results indicate that the vorticity - stream function methods have better accuracy and more rapid convergency. The primitive variable methods including SMAC, Spalding, Crocco, and Chorin methods have accuracy very sensitive to the convergency tolerance in the pressure iteration. If the tolerance is refined, computational time will excalate. The reason is the inadequate pressure iteration scheme. Therefore, the pressure are not in good agreement with those yielded from other methods. But accurate velocities can be obtained, because of the high accuracy in predicting the pressure gradients. Since the techniques were not optimized, the comparison was not definitive.

Recently, Cebeci et al (49) used problems of driven cavity and entrance flow in a channel to compare Spalding method, vorticity - stream function method (by ADI - SOR solution procedure) and stream function method (by biharmonic formulation). The test cases covered a Reynolds number in the range of 50 to 3200. The conclusion is that the biharmonic method is superior to the other two techniques. However, the main drawbacks are the large storage requirement and longer computer time for the matrix computation.

For the free surface flows, Yeung (50) surveyed the methodologies of stream function. Three major methods were reviewed: finite differences, finite elements, and boundary - intergral equations. The finite difference is used inevitably in transient problems. The finite element has more flexibility to deal with arbitrary boundary geometry with little loss of accuracy. This is the newest method of the above mentioned three. More storage is required and some difficulities as mentioned on page 16 have to be resolved. The integral - equation formulation is able to reduce the space dimension by one and requires less storage. However, there is no accurate and efficient means to integrate the Green function.

The most important criteria of numerical modeling are stability, accuracy, and convergence. The next is size of data-storage. Emmons (51) and Orszag and Israeli (52) discussed those criteria for numerical simulation and served as the excellent starting point in selection of method.

APPENDIX VIII

COMPUTER LISTING OF THE EXAMPLE

```
REQUESTED OPTIONS (EXECUTE): NODECK,NOLIST,NOXREF,NOMAP,NOFIPS,FIXED,OPT(0),LANGLVL(66)

OPTIONS IN EFFECT:  NOLIST NOMAP NOXREF   GOSTMT NODECK   SOURCE   TERM   OBJECT FIXED
                      OPTIMIZE(0)  LANGLVL(66) NOFIPS   FLAG(I)  NAME(MAIN  )  LINECOUNT(60)

               *....*...1.........2.........3.........4.........5.........6.........7.*.......8

               C          DATA SET SOLAS7     AT LEVEL 223 AS OF 82/05/13
               C
ISN      1           IMPLICIT REAL*8(A-H,O-Z)
ISN      2           DIMENSION U(140,35),V(140,35),UN(140,35),VN(140,35),UV(140,35),
                    1P(140,35),ANG(140,35),H(140),HN(140),JT(140),JTO(140),JTU(140),
                    2HB(140),HBN(140),JB(140),RLH(140),BL(140),JBL(70),BR(70),JBR(70),
                    3DR(70),SLOB(70),CTA(70),VSIN(140),VCOS(140),BU(140),BV(140),
                    4HTU(140),TU(140),TV(140),TVSIN(140),TVCOS(140),CODTB(140),
                    5CODBB(140),HBU(140),JBU(140),JBO(140),DCTA(70),DS(140),TTM(140)
ISN      3           REAL NU
ISN      4           INTEGER CYCLE,CODTB,CODBB,SIGNL,TTM
ISN      5           PRINT 35
               C
               C * * READ AND PRINT INITIAL INPUT DATA
               C
ISN      6           IBAR=64
ISN      7           JBAR=16
ISN      8           DELX=.2
ISN      9           DELY=.4
ISN     10           AR=JBAR*DELY
ISN     11           DELT=.0005
ISN     12           NU=2.
ISN     13           EPSI=.1
ISN     14           UPSI=.1
ISN     15           FLHT=3.5
ISN     16           UI=0.
ISN     17           VI=0.
ISN     18           AS=220.
ISN     19           DLPHA=.9
ISN     20           ULPHA=.9
ISN     21           GAMMA=.2
ISN     22           OMG=1.4
ISN     23           UMG=1.4
ISN     24           GX=0.
ISN     25           GY=-980.
ISN     26        25 FORMAT(7X,'TOP BOUNDARY CONDITION FAILED TO CONVERGE')
ISN     27        26 FORMAT(7X,'BOTTOM FREE SURFACE MODIFIED',2X,'I=',I3,3X,'JB1=',I3)
ISN     28        27 FORMAT(7X,'* * * BOTTOM SURFACE MODIFIED * * *')
ISN     29        29 FORMAT(10X,'ILL1=',I4,5X,'JT(ILL)=',I6,5X,'H(ILL)=',1PE12.5)
ISN     30        30 FORMAT(10X,'IR=',I4,5X,'JT(IR+1)=',I6,5X,'H(IR+1)=', 1PE12.5)
ISN     31        31 FORMAT(10X,'IRU=',I4,5X,'ILU=',I4,5X,'IRUT=',I4,5X,'TDVOL=',1PE12.
                    15,5X,'KASE=',I3)
ISN     32        32 FORMAT(10X,'IRUT=',I4,5X,'JB(IRUT)=',I6,5X,'HB(IRUT)=', 1PE12.5)
ISN     33        33 FORMAT(10X,'IRUT=',I4,5X,'JB(IRUT+1)=',I6,5X,'HB(IRUT+1)=',1PE12.5
                    1)
ISN     34        35 FORMAT(1H1)
ISN     35        38 FORMAT(2X,I3,'-',I2,2X,   I3,4(4X,1PE12.5),2(4X,I3),2(6X,1PE12.5))
ISN     36        39 FORMAT(6X,'ITER=',I5,10X,'TIME=',1PE12.5,10X,'SIGNL=',I4,10X,
                    1'VOLUME=',1PE12.5)
ISN     37        40 FORMAT(/10X,'TOP BOUNDARY WAS REACHED AT I=',I5)
ISN     38        41 FORMAT(/10X,'BOTTOM BOUNDARY WAS REACHED AT I=',I5)
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN   39         42 FORMAT(10X,'LR=',I4,5X,'JT(LR)=',I6,5X,'H(LR)=',1PE12.5)
ISN   40         47 FORMAT(4X,'I',7X,'J',10X,'U',15X,'V',15X,'P',15X,'H',6X,'SUR CELL'
                   1,3X,'BOT CELL',9X,'UV',14X,'ANG')
ISN   41         48 FORMAT(2X,I3,5X,I3,4(4X,1PE12.5),2(4X,I3),2(6X,1PE12.5))
ISN   42         49 FORMAT(6X,'ITER=',I5,10X,'TIME=',1PE12.5,10X,'CYCLE=',I4,10X,
                   1'VOLUME=',1PE12.5)
                C
                C * * COMPUTE CONSTANT TERMS AND INITIALIZE NECESSARY VARIABLES
                C
ISN   43            MEET=0
ISN   44            ILU=0
ISN   45            IRU=0
ISN   46            LL=0
ISN   47            LR=0
ISN   48            IMAX=IBAR+2
ISN   49            JMAX=JBAR+2
ISN   50            IM1=IMAX-1
ISN   51            JM1=JMAX-1
ISN   52            RDX=1.0/DELX
ISN   53            RDY=1.0/DELY
ISN   54            IM2=2*IM1
ISN   55            JM2=2*JM1
ISN   56            IMX=IMAX/2
ISN   57            T=0.
ISN   58            CWPRT=50
ISN   59            TWPRT=9.*DELT
ISN   60            SIGNL=0
ISN   61            TTU=0.
ISN   62            CUPRT=50.
ISN   63            TUPRT=17 .*DELT
ISN   64            BETA=OMG/(2.*DELT*(RDX**2+RDY**2))
ISN   65            UBETA=UMG/(2.*DELT*(RDX**2+RDY**2))
ISN   66            FVOU=0.
ISN   67            FLOL=0.
ISN   68            LYCLE=0
ISN   69            TTD=0.
ISN   70            CLPRT=10.
ISN   71            TLPRT=8  .*DELT
                C
                C * * DETREMINE BOTTOM BOUNDARY LOCATION
                C
ISN   72            BX=0.
ISN   73            CY=AR
ISN   74            BL(2)=CY*RDY
ISN   75            JBL(2)=BL(2)
ISN   76            DL=BL(2)-DFLOAT(JBL(2))
ISN   77            DO 190 I=2,IMX
ISN   78            BX=BX+DELX
ISN   79            BY=AR-DSQRT(2.*AR*BX-BX*BX)
ISN   80            BR(I)=BY*RDY
ISN   81            JBR(I)=BR(I)
ISN   82            DR(I)=BR(I)-DFLOAT(JBR(I))
ISN   83            NB=JBL(I)-JBR(I)
ISN   84            IF(NB .EQ. 0) GO TO 60
ISN   85            IF(NB .EQ. 1) GO TO 90
ISN   86            IF(NB .EQ. 2) GO TO 130
```

```
                  *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      87            GO TO 150
ISN      88         60 CB=DL+DR(I)
ISN      89            IF(CB .LT. 1.5) GO TO 150
ISN      90            JB(I)=JBL(I)+3
ISN      91            GO TO 160
ISN      92         90 BK=DL/(1.-DR(I))
ISN      93            RBX=1./(1.+BK)
ISN      94            CB=(1.-DR(I))/(1.+BK)
ISN      95            IF(CB .GE. .5) GO TO 150
ISN      96            JB(I)=JBR(I)+3
ISN      97            IF(RBX .GE. 0.5) GO TO 160
ISN      98            HB(I)=.5*(CY+BY)
ISN      99            GO TO 170
ISN     100        130 CONTINUE
ISN     101            JB(I)=JBR(I)+3
ISN     102            HB(I)=.5*(BY+CY)
ISN     103            GO TO 170
ISN     104        150 HB(I)=.5*(BY+CY)
ISN     105            JB(I)=    HB(I)*RDY+1.E-8+2.
ISN     106            GO TO 170
ISN     107        160 HB(I)=(FLOAT(JB(I))-2)*DELY
ISN     108        170 CONTINUE
ISN     109            SLOB(I)=   -(AR-(BX-.5*DELX))/(AR-HB(I))
ISN     110            SLOB(IMAX-I+1)=-SLOB(I)
ISN     111            CTA(I)=DATAN(SLOB(I))
ISN     112            CTA(IMAX-I+1)=-CTA(I)
ISN     113            HB(IMAX-I+1)=HB(I)
ISN     114            JB(IMAX-I+1)=JB(I)
ISN     115            HBN(I)=HB(I)
ISN     116            HBN(IMAX-I+1)=HBN(I)
ISN     117        180 CONTINUE
ISN     118            CY=BY
ISN     119            BL(I+1)=BR(I)
ISN     120            JBL(I+1)=JBR(I)
ISN     121            DL=DR(I)
ISN     122            DR(IMAX-I+1)=DR(I)
ISN     123        190 CONTINUE
ISN     124            HB(1)=HB(2)
ISN     125            HB(IMAX)=HB(IM1)
ISN     126            JB(1)=JB(2)
ISN     127            JB(IMAX)=JB(IM1)
ISN     128            CTA(1)=CTA(2)
ISN     129            CTA(IMAX)=CTA(IM1)
ISN     130            DO 200 I=2,IM1
ISN     131            VCOS(I)=DCOS(CTA(I))
ISN     132            VSIN(I)=DSIN(CTA(I))
ISN     133        200 CONTINUE
                  C
                  C * * COMPONENTS OF ROTATION VELOCITY
                  C
ISN     134            DO 210 I=2,IMX
ISN     135            BU(I)=AS*VCOS(I)
ISN     136            BV(I)=AS*VSIN(I)
ISN     137            BU(IMAX-I+1)=BU(I)
ISN     138            BV(IMAX-I+1)=-BV(I)
ISN     139        210 CONTINUE
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

                C
                C * * COMPUTE INITIAL TOP SURFACE CONFIGURATION
                C
ISN     140           DO 230 I=1,IMAX
ISN     141           TTM(I)=0.
ISN     142           H(I)=HB(I)
ISN     143           HN(I)=H(I)
ISN     144           JT(I)=JB(I)
ISN     145           RLH(I)=0.
ISN     146           CODTB(I)=0
ISN     147           CODBB(I)=0
ISN     148           RLH(I+IMAX)=0.
ISN     149           CODTB(I+IMAX)=0
ISN     150           CODBB(I+IMAX)=0
ISN     151       230 CONTINUE
ISN     152           TX=AR-DSQRT(2.*AR*FLHT-FLHT*FLHT)
ISN     153           IL=    TX*RDX+1.E-8+2.
ISN     154           JLHT =    FLHT*RDY+1.E-8+1.
ISN     155       240 IF(JLHT .GE. JB(IL)) GO TO 250
ISN     156           IL=IL+1
ISN     157           GO TO 240
ISN     158       250 CONTINUE
ISN     159           IR=IMAX-IL+1
ISN     160           IL1=IL-1
ISN     161           IR1=IR+1
ISN     162           DO 260 I=IL1,IR1
ISN     163           H(I)=FLHT
ISN     164           HN(I)=H(I)
ISN     165           JT(I)=JLHT+1
ISN     166           RLH(I)=H(I)-HB(I)
ISN     167       260 CONTINUE
ISN     168           FFVOL=0.
ISN     169           DO 270 I=IL,IR
ISN     170           FFVOL=FFVOL+(H(I)-HB(I))*DELX
ISN     171       270 CONTINUE
                C
                C * * CALCULATE HYDROSTATIC PRESSURE
                C
ISN     172           DO 280 I=2,IM2
ISN     173           DO 280 J=1,JM2
ISN     174           P(I,J)=0.
ISN     175       280 CONTINUE
ISN     176           DO 290 I=IL1,IMX
ISN     177           JT1=JT(I)
ISN     178           JB1=JB(I)
ISN     179           DO 290 J=JB1,JT1
ISN     180           P(I,J)=-GY*(H(I)-(DFLOAT(J)-1.5)*DELY)
ISN     181           P(IMAX-I+1,J)=P(I,J)
ISN     182       290 CONTINUE
                C
                C * * SET INITIAL VELOCITY FIELD INTO U AND V ARRAYS
                C
ISN     183           DO 300 I=1,IM2
ISN     184           DO 300 J=1,JM2
ISN     185           U(I,J)= UI
ISN     186           V(I,J)=VI
```

```
                 *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     187        300 CONTINUE
ISN     188            DO 350 I=IL1,IR1
ISN     189            JB1=JB(I)
ISN     190            U(I,JB1)=BU(I)
ISN     191            V(I,JB1)=BV(I)
ISN     192        350 CONTINUE
ISN     193            DO 400 I=IL,IR
ISN     194            CODBB(I)=1
ISN     195            JB1=JB(I)
ISN     196            IF(U(I-1,JB1) .EQ. 0) U(I-1,JB1)=U(I,JB1)
ISN     198            V(I,JB1-1)=V(I,JB1)+DELY*RDX*(U(I,JB1)-U(I-1,JB1))
ISN     199            U(I,JB1-1)=(AS-V(I,JB1-1)*VSIN(I))/VCOS(I)
ISN     200        400 CONTINUE
                 C
                 C * * SET RIGHT AND LEFT BOUNDARY BY SLOP
                 C
ISN     201            ALMT=DELY*RDX
ISN     202            I=2
ISN     203        420 IF(DABS(SLOB(I)) .LE. ALMT) GO TO 500
ISN     204            DO 450 J=2,JM2
ISN     205            U(I,J)=0.
ISN     206            V(I,J)=-AS
ISN     207            U(IMAX-I+1,J)=0.
ISN     208            V(IMAX-I+1,J)=AS
ISN     209            BU(I)=0.
ISN     210            BV(I)=-AS
ISN     211            BU(IMAX-I+1)=0.
ISN     212            BV(IMAX-I+1)=AS
ISN     213        450 CONTINUE
ISN     214            I=I+1
ISN     215            GO TO 420
ISN     216        500 CONTINUE
ISN     217            ML=I
ISN     218            MR=IMAX-I+1
ISN     219            BU(MR)=0.
ISN     220            BV(MR)=AS
ISN     221            MRU=2*IMAX - MR
ISN     222            MLU=2*IMAX - ML
ISN     223            DO 520 J=2,JM2
ISN     224            U(MR,J)=0.
ISN     225            V(MR,J)=AS
ISN     226            U(MLU,J)=0.
ISN     227            V(MLU,J)=-AS
ISN     228        520 CONTINUE
                 C
                 C * * TOP BOUNDARY FOR LOWER HALF PORTION
                 C
ISN     229            ML1=ML-1
ISN     230            MR1=MR+1
ISN     231            DO 550 I=ML1,MR1
ISN     232            JTU(I)=JM1+JMAX-JB(I)
ISN     233            HTU(I)=2.*AR -HB(I)
ISN     234            TU(I)=-BU(I)
ISN     235            TV(I)=BV(I)
ISN     236            TVSIN(I)=VSIN(I)
ISN     237            TVCOS(I)=-VCOS(I)
```

```
                 *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     238        550 CONTINUE
ISN     239            JB1=JB(MR)
ISN     240            JT1=JTU(MR)-1
ISN     241            DO 580 J=JB1,JT1
ISN     242            U(MR,J)=0.
ISN     243            UN(MR,J)=0.
ISN     244            U(MR+1,J)=0.
ISN     245            UN(MR+1,J)=0.
ISN     246        580 CONTINUE
                 C
                 C * * TOP BOUNDARY FOR UPPER HALF PORTION
                 C
ISN     247            DO 600 I=ML1,MR1
ISN     248            IU=2*IMAX-I
ISN     249            TU(IU)=BU(I)
ISN     250            TV(IU)=BV(I)
ISN     251            TVCOS(IU)=VCOS(I)
ISN     252            TVSIN(IU)=VSIN(I)
ISN     253            JTU(IU)=JTU(I)
ISN     254            HTU(IU)=HTU(I)
ISN     255            HBU(IU)=HB(I)
ISN     256            JBU(IU)=JB(I)
ISN     257            JT(IU)=JTU(IU)
ISN     258            H(IU)=HTU(IU)
ISN     259            JB(IU)=JT(IU)
ISN     260            HB(IU)=H(IU)
ISN     261            HBN(IU)=HB(IU)
ISN     262        600 CONTINUE
                 C
                 C * * INITIAL VELOCITIES AND PRESSURE IN UPPER HALF
                 C
ISN     263            DO 610 I=IL1,IM1
ISN     264            IU=2*IMAX-I
ISN     265            DO 610 J=2,JM2
ISN     266            U(IU,J)=0.
ISN     267            V(IU,J)=0.
ISN     268            P(IU,J)=0.
ISN     269        610 CONTINUE
                 C
                 C * * BOTTOM BOUNDARY CONDITIONS
                 C
ISN     270            ILS=IL
ISN     271            IRS=IR
ISN     272            MRS=MR
ISN     273            ITB=0
ISN    •274        650 CONTINUE
ISN     275            BFLG=0.
ISN     276            ITB=ITB+1
ISN     277            IF(ITB  .GE. 51) GO TO 670
ISN     278            DO 660  I=ILS,IRS
ISN     279            J=JB(I)
ISN     280            VTM=RDY*(HB(I)-(J-2)*DELY)
ISN     281            VBM=RDY*((J-1)*DELY-HB(I))
ISN     282            F=-0.25*RDX*(HB(I+1)-HB(I-1))*(U(I,J)+U(I-1,J))+V(I,J)*VTM
                      1+VBM*(V(I,J)+DELY*RDX*(U(I,J)-U(I-1,J)))
ISN     283            DFDP=DELT*RDY*(1.0+2.0*VBM*DELY**2*RDX**2)
```

```
                 *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      284           DELP=-F/DFDP
ISN      285           CORP=DELP/P(I,J)
ISN      286           IF(DABS(CORP) .GE. .05) BFLG=1.
ISN      288           P(I,J)=P(I,J)+DELP
ISN      289           U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN      290           V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN      291           V(I,J)=V(I,J)+DELT*RDY*DELP
ISN      292           U(I,J)=U(I,J)+DELT*RDX*DELP
ISN      293       660 CONTINUE
ISN      294           IF(BFLG .EQ. 0.) GO TO 670
ISN      295           GO TO 650
ISN      296       670 CONTINUE
                 C
                 C * * SET BOUNDARY CONDITIONS
                 C
ISN      297           HN(IL-1)=HN(IL)
ISN      298           JT(IL-1)=JT(IL)
ISN      299           HN(IR+1)=HN(IR)
ISN      300           JT(IR+1)=JT(IR)
ISN      301           JB2=JB(IL)-1
ISN      302           JT1=JT(IL)+1
ISN      303           DO 700 J=JB2,JT1
ISN      304           U(IL-1,J)=U(IL,J)
ISN      305           V(IL-1,J)=V(IL,J)
ISN      306       700 CONTINUE
ISN      307           JT1=JT(IR)+1
ISN      308           JB2=JB(IR)-1
ISN      309           DO 750 J=JB2,JT1
ISN      310           U(IR+1,J)=U(IR,J)
ISN      311           V(IR+1,J)=V(IR,J)
ISN      312       750 CONTINUE
ISN      313           DO  800 I=IL1,IR1
ISN      314           JT2=JT(I)+1
ISN      315           JB2=JB(I)-1
ISN      316           DO  800 J=JB2,JT2
ISN      317           UN(I,J)=U(I,J)
ISN      318           VN(I,J)=V(I,J)
ISN      319           HN(I)=H(I)
ISN      320       800 CONTINUE
                 C
                 C * * START CYCLE
                 C
ISN      321           DO 20300 CYCLE=1,561
ISN      322           FLG=1.
ISN      323           ALPHA=DLPHA
                 C
                 C * * COMPUTE TEMPORARY U AND V
                 C
ISN      324           DO 1100 I=IL,IR
ISN      325           JT1= JT(I)
ISN      326           JT11=JT1-1
ISN      327           IF(CODTB(I) .EQ. 1) JT1=JT11
ISN      329           JB1=JB(I)+1
ISN      330           DO 1100 J=JB1,JT1
ISN      331           IF(I .EQ. MRS) GO TO 1050
ISN      332           FUX=((UN(I,J)+UN(I+1,J))*(UN(I,J)+UN(I+1,J))+ALPHA*DABS(UN(I,J)
```

```
                   *....*...1.........2.........3.........4.........5.........6.........7.*.......8

                        1+UN(I+1,J))*(UN(I,J)-UN(I+1,J))-(UN(I-1,J)+UN(I,J))*(UN(I-1,J)
                        2+UN(I,J))-ALPHA*DABS(UN(I-1,J)+UN(I,J))*(UN(I-1,J)-UN(I,J)))
                        3/(4.0*DELX)
ISN      333             FUY=((VN(I,J)+VN(I+1,J))*(UN(I,J)+UN(I,J+1))
                        1+ALPHA*DABS(VN(I,J)+VN(I+1,J))*(UN(I,J)-UN(I,J+1))
                        2-(VN(I,J-1)+VN(I+1,J-1))*(UN(I,J-1)+UN(I,J))
                        3-ALPHA*DABS(VN(I,J-1)+VN(I+1,J-1))*(UN(I,J-1)-UN(I,J)))/(4.*DELY)
ISN      334             VISX=NU*((UN(I+1,J)-2.*UN(I,J)+UN(I-1,J))/DELX**2+
                        1(UN(I,J+1)-2.*UN(I,J)+UN(I,J-1))/DELY**2)
ISN      335             U(I,J)=UN(I,J)+DELT*((P(I,J)-P(I+1,J))*RDX+GX-FUX-FUY     +VISX)
ISN      336       1050 CONTINUE
ISN      337             FVX=((UN(I,J)+UN(I,J+1))*(VN(I,J)+VN(I+1,J))+ALPHA*DABS(UN(I,J)
                        1+UN(I,J+1))*(VN(I,J)-VN(I+1,J))-(UN(I-1,J)+UN(I-1,J+1))*(VN(I-1,J)
                        2+VN(I,J))-ALPHA*DABS(UN(I-1,J)+UN(I-1,J+1))*(VN(I-1,J)-VN(I,J)))
                        3/(4.0*DELX)
ISN      338             FVY=((VN(I,J)+VN(I,J+1))*(VN(I,J)+VN(I,J+1))+ALPHA*DABS(VN(I,J)+VN
                        1(I,J+1))*(VN(I,J)-VN(I,J+1))-(VN(I,J-1)+VN(I,J))*(VN(I,J-1)+VN(I,J
                        2))-ALPHA*DABS(VN(I,J-1)+VN(I,J))*(VN(I,J-1)-VN(I,J)))/(4.*DELY)
ISN      339             VISY=NU*((VN(I+1,J)-2.*VN(I,J)+VN(I-1,J))/DELX**2+
                        1(VN(I,J+1)-2.*VN(I,J)+VN(I,J-1))/DELY**2)
ISN      340             V(I,J)=VN(I,J)+DELT*((P(I,J)-P(I,J+1))*RDY+GY-FVX-FVY     +VISY)
ISN      341       1100 CONTINUE
                   C
                   C * * BOTTOM BOUNDARY CONDITIONS
                   C
ISN      342             ITB=0
ISN      343       1250 CONTINUE
ISN      344             BFLG=0.
ISN      345             ITB=ITB+1
ISN      346             IF(ITB  .GE. 51) GO TO 1400
ISN      347             DO 1300 I=ILS,IRS
ISN      348             J=JB(I)
ISN      349             VTM=RDY*(HB(I)-(J-2)*DELY)
ISN      350             VBM=RDY*((J-1)*DELY-HB(I))
ISN      351             F=-0.25*RDX*(HB(I+1)-HB(I-1))*(U(I,J)+U(I-1,J))+V(I,J)*VTM
                        1+VBM*(V(I,J)+DELY*RDX*(U(I,J)-U(I-1,J)))
ISN      352             DFDP=DELT*RDY*(1.0+2.0*VBM*DELY**2*RDX**2)
ISN      353             DELP=-F/DFDP
ISN      354             CORP=DELP/P(I,J)
ISN      355             IF(DABS(CORP) .GE. .05) BFLG=1.
ISN      357             P(I,J)=P(I,J)+DELP
ISN      358             IF(I .EQ. LL) GO TO 1270
ISN      359             U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN      360       1270 CONTINUE
ISN      361             V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN      362             V(I,J)=V(I,J)+DELT*RDY*DELP
ISN      363             IF(I .EQ. MRS) GO TO 1300
ISN      364             U(I,J)=U(I,J)+DELT*RDX*DELP
ISN      365       1300 CONTINUE
ISN      366             IF(BFLG .EQ. 0.) GO TO 1400
ISN      367             GO TO 1250
ISN      368       1400 CONTINUE
                   C
                   C * * TOP RIGID BOUNDARY CONDITIONS
                   C
ISN      369             ITU=0
```

```
                  *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      370      1500 ITU=ITU+1
ISN      371           UFLG=0.
ISN      372           IF(ITU .GE. 50) GO TO 1700
ISN      373           DO 1600 I=ILS,IRS
ISN      374           IF(CODTB(I) .EQ. 0) GO TO 1600
ISN      375           J=JT(I)
ISN      376           VTM=RDY*(HTU(I)-(J-2)*DELY)
ISN      377           F=-.25*RDX*(HTU(I+1)-HTU(I-1))*(U(I,J)+U(I-1,J)) +V(I,J-1)
                      1-VTM*DELY*RDX*(U(I,J)-U(I-1,J))
ISN      378           DFDP=-DELT*RDY*(1.+2.*VTM*DELY**2*RDX**2)
ISN      379           DELP=-F/DFDP
ISN      380           CORP=DELP/P(I,J)
ISN      381           IF(DABS(CORP) .GE. .05) UFLG=1.
ISN      383           P(I,J)=P(I,J)+DELP
ISN      384           V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN      385           V(I,J)=V(I,J)+DELT*RDY*DELP
ISN      386           U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN      387           IF(I .EQ. MRS) GO TO 1600
ISN      388           U(I,J)=U(I,J)+DELT*RDX*DELP
ISN      389      1600 CONTINUE
ISN      390           IF(UFLG .EQ. 0.) GO TO 1700
ISN      391           GO TO 1500
ISN      392      1700 CONTINUE
                  C
                  C * * HAS CONVERGENCE BEEN REACHED
                  C
ISN      393           DO 3600 ITER=1,200
                  C
                  C * * SET LEFT AND RIGHT BOUNDARY CONDITIONS
                  C
ISN      394           JB2=JB(IR)-1
ISN      395           JT2=JT(IR)+1
ISN      396           IF(IR .LT. MR) GO TO 1810
ISN      397           DO 1800 J=JB2,JT2
ISN      398           V(IR+1,J)=V(IR,J)*2. -V(IR-1,J)
ISN      399      1800 CONTINUE
ISN      400           GO TO 1855
ISN      401      1810 CONTINUE
ISN      402           DO 1850 J=JB2,JT2
ISN      403           IF(J .EQ. JB(IR+1)) GO TO 1850
ISN      404           IF(IR+1 .EQ. MR) GO TO 1850
ISN      405           V(IR+1,J)=V(IR,J)
ISN      406           U(IR+1,J)=U(IR,J)
ISN      407      1850 CONTINUE
ISN      408      1855 CONTINUE
ISN      409           U(IR+1,JB2+1)=BU(IR+1)
ISN      410           V(IR+1,JB2+1)=BV(IR+1)
                  C
                  C * * FREE SURFACE CONDITIONS
                  C
ISN      411           DO 2620 I=IL,IR
ISN      412           IF(H(I) .GE. HTU(I)) GO TO 2620
ISN      413           JT1=JT(I)
ISN      414           IF(I .LE. 6) GO TO 2600
ISN      415           IF(JT(I+1).LT.JT(I)) U(I,JT1)=U(I,JT1-1)
ISN      417      2600 CONTINUE
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      418          V(I,JT1)=V(I,JT1-1)-DELY*RDX*(U(I,JT1)-U(I-1,JT1))
ISN      419          U(I,JT1+1)=U(I,JT1)
ISN      420     2620 CONTINUE
                C
                C * * COMPUTE UPDATED CELL PRESSURE AND VELOCITY
                C
ISN      421     3050 FLG=0.0
ISN      422          DO 3500 I=IL,IR
ISN      423          D=0.
ISN      424          JT1=JT(I)
ISN      425          JT11=JT(I)-1
ISN      426          IF(CODTB(I) .EQ. 1) JT1=JT11
ISN      428          JB11=JB(I)+1
ISN      429          DO 3500 J=JB11,JT1
ISN      430          IF(J .NE. JT1) GO TO 3200
ISN      431          IF(CODTB(I) .EQ. 1) GO TO 3500
ISN      432          PETA=DELY/(HN(I)-(DFLOAT(JT1)-2.5)*DELY)
ISN      433          DELP=(1.0-PETA)*P(I,JT1-1)-P(I,JT1)
ISN      434          GO TO 3300
ISN      435     3200 CONTINUE
ISN      436          D=RDX*(U(I,J)-U(I-1,J))+RDY*(V(I,J)-V(I,J-1))
ISN      437          IF(DABS(D)      .GE.EPSI)FLG=1.0
ISN      439          DELP=-BETA*D
ISN      440     3300 P(I,J)=P(I,J)+DELP
ISN      441          IF(I .EQ. MR) GO TO 3400
ISN      442          U(I,J)=U(I,J)+DELT*RDX*DELP
ISN      443     3400 CONTINUE
ISN      444          IF(I .EQ. LL) GO TO 3450
ISN      445          U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN      446     3450 CONTINUE
ISN      447          V(I,J)=V(I,J)+DELT*RDY*DELP
ISN      448          IF(J .EQ. JB11) GO TO 3500
ISN      449          V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN      450     3500 CONTINUE
ISN      451          IF(FLG .EQ. 0.) GO TO 4000
ISN      452     3600 CONTINUE
                C
                C * * COMPUTE NEW SURFACE POSITION
                C
ISN      453     4000 CONTINUE
ISN      454          DO 4100 I=IL,IR
ISN      455          IF(I .EQ. MR) GO TO 4050
ISN      456          JT1=JT(I)
ISN      457          HV=RDY*(HN(I)-DFLOAT(JT1-2)*DELY)
ISN      458          UAV=0.5*(U(I-1,JT1) +U(I,JT1))
ISN      459          H(I)=HN(I)+DELT*(HV*V(I,JT1)+(1.0-HV)*V(I,JT1-1)
                     1     -0.5*RDX*(UAV*HN(I+1)+GAMMA*DABS(UAV)*(HN(I)-HN(I+1))
                     2     -UAV*HN(I-1)-GAMMA*DABS(UAV)*(HN(I-1)-HN(I))))
ISN      460          IF(H(I) .LT. HB(I)) H(I)=HB(I)
ISN      462          GO TO 4080
ISN      463     4050 CONTINUE
ISN      464          H(I)=HN(I)+AS*DELT
ISN      465     4080 CONTINUE
ISN      466          IF(H(I) .GT. HTU(I)) H(I)=HTU(I)
ISN      468          RLH(I)=H(I)-HB(I)
ISN      469     4100 CONTINUE
```

```
                 *....*...1.........2.........3.........4.........5.........6.........7.*.......8

                 C
                 C * * CALCULATE CELL IN WHICH SURFACE IS LOCATED AND UPDATE ARRAY
                 C
ISN      470           DO 4250 I=IL,IR
ISN      471           JTO(I)=JT(I)
ISN      472           JT(I)=    H(I)*RDY+1.0E-8+2.
ISN      473           IF(JT(I) .GT. JTU(I)) JT(I)=JTU(I)
ISN      475           IF(JT(I) .EQ. JTO(I)) GO TO 4250
ISN      476           JT1=JT(I)
ISN      477           IF(JT1 .GT. JTO(I)) GO TO 4240
ISN      478           J1=JT1+1
ISN      479           JTO1=JTO(I)
ISN      480           DO 4230 J=J1,JTO1
ISN      481           U(I,J)=0.
ISN      482           V(I,J)=0.
ISN      483           P(I,J)=0.
ISN      484      4230 CONTINUE
ISN      485           GO TO 4250
ISN      486      4240 JT11=JTO(I)+1
ISN      487           DO 4250 J=JT11,JT1
ISN      488           P(I,J)=-GY*(H(I)-(DFLOAT(J)-1.5)*DELY)
ISN      489      4250 CONTINUE
                 C
                 C * * CHECK REACH THE TOP SURFACE
                 C
ISN      490           ILUB=0
ISN      491           DO 4270 I=IL,IR
ISN      492           CODBB(I)=0
ISN      493           CODTB(I)=0
ISN      494           IF(H(I) .NE. HTU(I)) GO TO 4270
ISN      495           IF(ILUB .NE. 0) GO TO 4260
ISN      496           ILUB=2*IMAX - I
ISN      497           HBN(ILUB)=HBN(I)
ISN      498           JB2=JB(I)-1
ISN      499           JT2=JT(I)+1
ISN      500           DO 4260 J=JB2,JT2
ISN      501           U(ILUB-1,J)=-U(I,J)
ISN      502           U(ILUB,J)=-U(I-1,J)
ISN      503           V(ILUB,J)= V(I,J)
ISN      504           V(ILUB-1,J)=V(I+1,J)
ISN      505           P(ILUB,J)=P(I,J)
ISN      506      4260 CONTINUE
ISN      507           IU=2*IMAX-I
ISN      508           CODBB(IU)=1
ISN      509           CODTB(IU)=1
ISN      510           CODTB(I)=1
ISN      511           JT1=JT(I)
ISN      512           U(I,JT1)=TU(I)
ISN      513           V(I,JT1)=TV(I)
ISN      514           U(I,JT1+1)=U(I,JT1)
ISN      515           V(I,JT1+1)=V(I,JT1)
ISN      516      4270 CONTINUE
                 C
                 C * * NEW IL LOCATION
                 C
ISN      517           IF(MEET .EQ. 0) GO TO 4550
```

```
                 *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN    518            IL8=IL
ISN    519            GO TO 4650
ISN    520       4550 CONTINUE
ISN    521            IL8=IL+4
ISN    522            K=IL
ISN    523            DO 4600 I=K,IL8
ISN    524            IF(JT(I) .GT. JB(I)) GO TO 4600
ISN    525            CODBB(I)=0
ISN    526            IL=I+1
ISN    527       4600 CONTINUE
ISN    528       4650 CONTINUE
ISN    529            IR2=IR-1
ISN    530            DO 4700 I=IL8,IR2
ISN    531            IF(JT(I) .GT. JB(I)) GO TO 4700
ISN    532            JT(I)=JB(I)+1
ISN    533            J=JT(I)
ISN    534            H(I)=(DFLOAT(J)-2.)*DELY
ISN    535            RLH(I)=H(I)-HB(I)
ISN    536            V(I,J)=V(I+1,J)
ISN    537            U(I,J)=U(I-1,J)+(V(I,J-1)-V(I,J))*DELX*RDY
ISN    538            P(I,J)=-GY*(H(I)-(DFLOAT(J)-1.5)*DELY)
ISN    539       4700 CONTINUE
ISN    540            IL1=IL-1
                 C
                 C * * CALCULATE NEW VOLUME
                 C
ISN    541            FVOL=0.
ISN    542            DO 4280 I=IL,IR
ISN    543            FVOL=FVOL+(H(I)-HB(I))*DELX
ISN    544       4280 CONTINUE
ISN    545            IF(FFVOL .LE. FVOL) GO TO 4400
                 C
                 C * * NEW IR LOCATION
                 C
ISN    546       4300 CONTINUE
ISN    547            IF (IR .EQ. MR) GO TO 4400
ISN    548            NIR=JT(IR)-JB(IR)
ISN    549            IF(NIR .LE. 0) GO TO 4350
ISN    550            NIR=JT(IR)-JB(IR+1)
ISN    551            IF(NIR .LT. 1) GO TO 4400
ISN    552            H(IR+1)=HB(IR+1)+(FFVOL-FVOL)*RDX
ISN    553            JT(IR+1)=    H(IR+1)*RDY+2.
ISN    554            IF(JT(IR+1)-JB(IR+1) .LT. 1) GO TO 4400
ISN    555            PRINT 30,IR,JT(IR+1),H(IR+1)
ISN    556            IR=IR+1
ISN    557            CODBB(IR)=1
ISN    558            JT1=JT(IR)
ISN    559            JB1=JB(IR)
ISN    560            DO 4320 J=JB1,JT1
ISN    561            P(IR,J)=P(IR-1,J)
ISN    562            IF (IR .EQ. MR) GO TO 4320
ISN    563            U(IR,J)=U(IR-1,J)
ISN    564            V(IR,J)=V(IR-1,J)
ISN    565       4320 CONTINUE
ISN    566            U(IR,JB1)=BU(IR)
ISN    567            V(IR,JB1)=BV(IR)
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      568           RLH(IR)=H(IR)-HB(IR)
ISN      569           V(IR,JB1-1)=V(IR,JB1)+(U(IR,JB1)-U(IR-1,JB1 ))*DELY*RDX
ISN      570           U(IR,JB1-1)=(AS-V(IR,JB1-1)*VSIN(IR))/VCOS(IR)
ISN      571           GO TO 4400
ISN      572      4350 IR=IR-1
ISN      573           CODBB(IR+1)=0
ISN      574           NIR=JT(IR)-JB(IR)
ISN      575           IF(NIR .LT. 1) GO TO 4350
ISN      576      4400 CONTINUE
ISN      577           IF(IR .EQ. MR) GO TO 4500
ISN      578           IR1=IR+1
ISN      579           JB1=JB(IR1)
ISN      580           U(IR1,JB1)=BU(IR1)
ISN      581           V(IR1,JB1)=BV(IR1)
ISN      582           P(IR1,JB1)=P(IR,JB1)
                C
                C * * BOTTOM BOUNDARY CONDITIONS
                C
ISN      583      4500 CONTINUE
ISN      584           MRS=MR
ISN      585           IRS=IR
ISN      586           ILS=IL
ISN      587           ITB=0
ISN      588      4710 CONTINUE
ISN      589           BFLG=0.
ISN      590           ITB=ITB+1
ISN      591           IF(ITB  .GE. 51) GO TO 4730
ISN      592           DO 4720 I=ILS,IRS
ISN      593           J=JB(I)
ISN      594           VTM=RDY*(HB(I)-(J-2)*DELY)
ISN      595           VBM=RDY*((J-1)*DELY-HB(I))
ISN      596           F=-0.25*RDX*(HB(I+1)-HB(I-1))*(U(I,J)+U(I-1,J))+V(I,J)*VTM
                     1+VBM*(V(I,J)+DELY*RDX*(U(I,J)-U(I-1,J)))
ISN      597           DFDP=DELT*RDY*(1.0+2.0*VBM*DELY**2*RDX**2)
ISN      598           DELP=-F/DFDP
ISN      599           CORP=DELP/P(I,J)
ISN      600           IF(DABS(CORP) .GE. .05) BFLG=1.
ISN      602           P(I,J)=P(I,J)+DELP
ISN      603           IF(I .EQ. LL) GO TO 4715
ISN      604           U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN      605      4715 CONTINUE
ISN      606           V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN      607           V(I,J)=V(I,J)+DELT*RDY*DELP
ISN      608           IF(I .EQ. MRS) GO TO 4720
ISN      609           U(I,J)=U(I,J)+DELT*RDX*DELP
ISN      610      4720 CONTINUE
ISN      611           IF(BFLG .EQ. 0.) GO TO 4730
ISN      612           GO TO 4710
ISN      613      4730 CONTINUE
                C
                C * * TOP BOUNDARY CONDITIONS
                C
ISN      614           ILS=IL
ISN      615           IRS=IR
ISN      616           MRS=MR
ISN      617           ITU=0
```

```
               *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN    618      4750 ITU=ITU+1
ISN    619           UFLG=0.
ISN    620           IF(ITU .GE. 50) GO TO 4770
ISN    621           DO 4760 I=ILS,IRS
ISN    622           IF(CODTB(I) .EQ. 0) GO TO 4760
ISN    623           J=JT(I)
ISN    624           VTM=RDY*(HTU(I)-(J-2)*DELY)
ISN    625           F=-.25*RDX*(HTU(I+1)-HTU(I-1))*(U(I,J)+U(I-1,J)) +V(I,J-1)
                    1-VTM*DELY*RDX*(U(I,J)-U(I-1,J))
ISN    626           DFDP=-DELT*RDY*(1.+2.*VTM*DELY**2*RDX**2)
ISN    627           DELP=-F/DFDP
ISN    628           CORP=DELP/P(I,J)
ISN    629           IF(DABS(CORP) .GE. .05) UFLG=1.
ISN    631           P(I,J)=P(I,J)+DELP
ISN    632           V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN    633           V(I,J)=V(I,J)+DELT*RDY*DELP
ISN    634           U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN    635           IF(I .EQ. MRS) GO TO 4760
ISN    636           U(I,J)=U(I,J)+DELT*RDX*DELP
ISN    637      4760 CONTINUE
ISN    638           IF(UFLG .EQ. 0.) GO TO 4770
ISN    639           GO TO 4750
               C
               C * * IR VELOCITIES
               C
ISN    640      4770 CONTINUE
ISN    641           JB2=JB(IR)-1
ISN    642           JT2=JT(IR)+1
ISN    643           IF(IR  .LT. MR) GO TO 4790
ISN    644           DO 4780 J=JB2,JT2
ISN    645           V(IR+1,J)=V(IR,J)*2. -V(IR-1,J)
ISN    646      4780 CONTINUE
ISN    647           GO TO 4798
ISN    648      4790 CONTINUE
ISN    649           DO 4795 J=JB2,JT2
ISN    650           IF(J .EQ. JB(IR+1)) GO TO 4795
ISN    651           V(IR+1,J)=V(IR,J)
ISN    652           IF(IR+1 .EQ. MR) GO TO 4795
ISN    653           U(IR+1,J)=U(IR,J)
ISN    654      4795 CONTINUE
ISN    655           V(IR+1,JB2+1)=BV(IR+1)
ISN    656           U(IR+1,JB2+1)=BU(IR+1)
ISN    657      4798 CONTINUE
               C
               C * * FREE SURFACE CONDITION
               C
ISN    658           DO 4800 I=IL,IR
ISN    659           IF(H(I) .GE. HTU(I)) GO TO 4800
ISN    660           JT1=JT(I)
ISN    661           IF(I .LE. 6) GO TO 4799
ISN    662           IF(JT(I+1).LT.JT(I)) U(I,JT1)=U(I,JT1-1)
ISN    664      4799 CONTINUE
ISN    665           V(I,JT1)=V(I,JT1-1)-DELY*RDX*(U(I,JT1)-U(I-1,JT1))
ISN   •666           U(I,JT1+1)=U(I,JT1)
ISN    667      4800 CONTINUE
               C
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

                C * * CALCULATE TOTAL VOLUME
                C
ISN     668      5000 CONTINUE
ISN     669           FVOL=0.
ISN     670           DO 5100 I=IL,IR
ISN     671           CODBB(I)=1
ISN     672           FVOL=FVOL+(H(I)-HB(I))*DELX
ISN     673      5100 CONTINUE
ISN     674           PRINT 49,ITER,T,CYCLE,FVOL
                C
                C * * CALCULATE VELOCITY VECTOR
                C
ISN     675      5200 CONTINUE
ISN     676           IF(T+1.E-6 .LT. TWPRT) GO TO 6000
ISN     677           TWPRT=TWPRT+CWPRT*DELT
ISN     678           DO 5300 I=IL,IR
ISN     679           JB2=JB(I)-1
ISN     680           JT2=JT(I)+1
ISN     681           DO 5300 J=JB2,JT2
ISN     682           UV(I,J)=DSQRT(U(I,J)*U(I,J)+V(I,J)*V(I,J))
ISN     683           IF(DABS(U(I,J)) .LE. 1.E-6) U(I,J)=1.E-6
ISN     685           ANG(I,J)=57.3*DATAN(V(I,J)/U(I,J))
ISN     686           IF(U(I,J) .EQ. 1.E-6) U(I,J)=0.
ISN     688      5300 CONTINUE
                C
                C * * LIST VELOCITY, PRESSURE, AND SURFACE POSITION
                C
ISN     689           PRINT 35
ISN     690           PRINT 47
ISN     691           DO 5900 I=IL,IR
ISN     692           JT1=JT(I)
ISN     693           JT2=JT1+1
ISN     694           JB1=JB(I)
ISN     695           JB2=JB1-1
ISN     696           DO 5900 J=JB2,JT2
ISN     697           PRINT 48,I,J,U(I,J),V(I,J),P(I,J),  H(I),JT1,JB1,UV(I,J),ANG(I,J)
ISN     698      5900 CONTINUE
                C
                C * * SET THE ADVANCE TIME VELOCITIES U AND V INTO THE UN AND VN ARRAYS
                C * * AND THE ADVANCE TIME SURFACE HEIGHT H INTO THE HN ARRAY
                C
ISN     699      6000 CONTINUE
ISN     700           DO 6010 I=IL1,IR1
ISN     701           JT2=JT(I)+1
ISN     702           JB2=JB(I)-1
ISN     703           DO 6010 J=JB2,JT2
ISN     704           UN(I,J)=U(I,J)
ISN     705           VN(I,J)=V(I,J)
ISN     706           HN(I)=H(I)
ISN     707      6010 CONTINUE
                C
                C * * INPUT TO UPPER SECTION
                C
ISN     708           IF(ILUB .EQ. 0) GO TO 16200
ISN     709           TDVOL=FFVOL-FVOL-FVOU-FLOL
ISN     710           IRUT=ILUB+1
```

```
                  *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      711          IF(IRU .EQ. 0) GO TO 6020
ISN      712          IF(IRUT .LT. IRU) GO TO 6030
ISN      713          IF(IRUT .EQ. ILU) GO TO 6040
ISN      714          IF(IRUT .GT. ILU) GO TO 6050
ISN      715          KASE=5.
ISN      716          IRU=IRUT
ISN      717          PRINT 31,IRU,ILU,IRUT,TDVOL,KASE
ISN      718          GO TO 6060
ISN      719     6020 KASE=1.
ISN      720          GO TO 6060
ISN      721     6030 KASE=2.
ISN      722          GO TO 6060
ISN      723     6040 KASE=3.
ISN      724          FVOU=(H(ILU)-HB(ILU))*DELX
ISN      725          TDVOL=FFVOL-FVOL-FVOU
ISN      726          PRINT 31,IRU,ILU,IRUT,TDVOL,KASE
ISN      727          GO TO 6060
ISN      728     6050 KASE=4.
ISN      729          TDVOL=FFVOL-FVOL
ISN      730     6060 CONTINUE
ISN      731          GO TO (6105,7000,11000,6105,11000),KASE
ISN      732     6105 CONTINUE
ISN      733          PRINT 31,IRU,ILU,IRUT,TDVOL,KASE
ISN      734          IF(TDVOL .LE. 0.) GO TO 16200
ISN      735          HB(IRUT)=H(IRUT)-TDVOL*RDX
ISN      736          JB(IRUT)=     HB(IRUT)*RDY + 1.E-6+2.
ISN      737          PRINT 32,IRUT,JB(IRUT),HB(IRUT)
ISN      738          IF(HB(IRUT) .LE. H(2*IMAX-IRUT)) GO TO 6110
ISN      739          IF(JT(IRUT) .EQ. JB(IRUT)) GO TO 16200
ISN      740          IRU=IRUT
ISN      741          IRU1=IRU-1
ISN      742          GO TO 6120
ISN      743     6110 CONTINUE
ISN      744          IRU=IRUT
ISN      745          HB(IRU)=H(2*IMAX-IRU)
ISN      746          IRU1=IRU-1
ISN      747          JB(IRU)=JT(2*IMAX-IRU)
ISN      748     6120 CONTINUE
ISN      749          HBN(IRU)=HB(IRU)
ISN      750          ILU=IRU
ISN      751          ILU1=ILU+1
ISN      752          JT1=JT(IRU)
ISN      753          JB1=JB(IRU)
ISN      754          IF(ILUB .NE. 2*IMAX-MR) GO TO 6150
ISN      755          DO 6130 J=JB1,JT1
ISN      756          U(IRU,J)=U(ILUB,J)
ISN      757          V(IRU,J)=V(ILUB,J)
ISN      758          P(IRU,J)=GY*((DFLOAT(J)-1.5)*DELY-HB(IRU))
ISN      759     6130 CONTINUE
ISN      760          U(IRU,JT1)=TU(IRU)
ISN      761          V(IRU,JT1)=TV(IRU)
ISN      762          CODTB(IRU)=1
ISN      763          GO TO 15000
ISN      764     6150 CONTINUE
ISN      765          JT1=JT(IRU)
ISN      766          JB1=JB(IRU)
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      767           DO 6170 J=JB1,JT1
ISN      768           J1=J+JT(IRU-1)-JT(IRU)
ISN      769           J2=J+JT(IRU-2)-JT(IRU)
ISN      770           U(IRU,J)=2.*U(IRU-1,J1 )-U(IRU-2,J2 )
ISN      771           V(IRU,J)=2.*V(IRU-1,J1 )-V(IRU-2,J2 )
ISN      772           P(IRU,J)=GY*((DFLOAT(J)-1.5)*DELY-HB(IRU))
ISN      773      6170 CONTINUE
ISN      774           CODTB(IRU)=1
ISN      775           U(IRU,JT1)=TU(IRU)
ISN      776           V(IRU,JT1)=TV(IRU)
                C
                C * * TOP BOUNDARY CONDITIONS
                C
ISN      777      6250 CONTINUE
ISN      778           ITU=0
ISN      779      6300 ITU=ITU+1
ISN      780           UFLG=0.
ISN      781           IF(ITU .GE. 500) GO TO 6400
ISN      782           DO 6350 I=IRU,ILU
ISN      783           J=JT(I)
ISN      784           VTM=RDY*(HTU(I)-(J-2)*DELY)
ISN      785           IF(DABS(U(I-1,J)) .LE. 1.E-6) U(I-1,J)=U(I,J)
ISN      787           F=-0.25*RDX*(H(I+1)-H(I-1))*(U(I,J)+U(I-1,J))+V(I,J-1)
                     1   -VTM*DELY*RDX*(U(I,J)-U(I-1,J))
ISN      788           DFDP= -DELT*RDY*(1.0+2.0*VTM*DELY**2*RDX**2
                     1     +0.25*RDX**2*DELY*(H(I+1)-H(I-1))*(AMAX0(0,JT(I)-JT(I-1))
                     2      -AMAX0(0,JT(I)-JT(I+1))))
ISN      789           DELP=-F/DFDP
ISN      790           IF(P(I,J) .EQ. 0.) P(I,J)=P(I,J-1)
ISN      792           IF(P(I,J) .EQ. 0.) GO TO 6350
ISN      793           CORP=DELP/P(I,J)
ISN      794           IF(DABS(CORP) .GE. 0.05 .AND. DABS(DELP) .GE. 200.) UFLG=1.
ISN      796           P(I,J)=P(I,J)+DELP
ISN      797           V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN      798           V(I,J)=V(I,J)+DELT*RDY*DELP
ISN      799           U(I,J)=U(I,J)+DELT*RDX*DELP
ISN      800           IF(I .EQ. IRU) GO TO 6350
ISN      801           U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN      802      6350 CONTINUE
ISN      803           IF(UFLG .EQ. 0.) GO TO 6400
ISN      804           GO TO 6300
ISN      805      6400 CONTINUE
ISN      806           IF(ITU .GE. 500) GO TO 6410
ISN      807           GO TO 6420
ISN      808      6410 PRINT 25
ISN      809      6420 CONTINUE
                C
                C
ISN      810           DO 6450 I=IRU,ILU
ISN      811           JT1=JT(I)
ISN      812           JT2=JT1+1
ISN      813           U(I,JT2)=U(I,JT1)
ISN      814           V(I,JT2)=V(I,JT1)
ISN      815      6450 CONTINUE
                C
                C * * BOTTOM SURFACE CONDITIONS  --  FREE SURFACE
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8
                C
ISN     816      6470 CONTINUE
ISN     817           DO 6500 I=IRU,ILU
ISN     818           JB1=JB(I)
ISN     819           V(I,JB1-1)=(V(I,JB1)+V(I,JB1-2)+V(I-1,JB1-1)+V(I+1,JB1-1))*0.25
ISN     820           IF(I .EQ. MLU) GO TO 6500
ISN     821           U(I,JB1-1)=(U(I,JB1)+U(I,JB1-2)+U(I-1,JB1-1)+U(I+1,JB1-1))*0.25
ISN     822      6500 CONTINUE
                C
                C
ISN     823           JT1=JT(ILU)+1
ISN     824           JB1=JB(ILU)-1
ISN     825           DO 6550 J=JB1,JT1
ISN     826           U(ILU1,J)=U(ILU,J)
ISN     827           V(ILU1,J)=V(ILU,J)
ISN     828      6550 CONTINUE
ISN     829           JT1=JT(ILU+1)
ISN     830           U(ILU1,JT1)=TU(ILU1)
ISN     831           V(ILU1,JT1)=TV(ILU1)
ISN     832           IF(KASE .EQ. 3) GO TO 6590
ISN     833           GO TO 15000
ISN     834      6590 CONTINUE
ISN     835           DO 6600 I=IRU,ILU1
ISN     836           JB2=JB(I)-1
ISN     837           JT2=JT(I)+1
ISN     838           DO  6600 J=JB2,JT2
ISN     839           UN(I,J)=U(I,J)
ISN     840           VN(I,J)=V(I,J)
ISN     841      6600 CONTINUE
ISN     842           GO TO 11000
                C
                C
ISN     843      7000 CONTINUE
ISN     844           PRINT 31,IRU,ILU,IRUT,TDVOL,KASE
ISN     845           DO 7100 I=IRUT,IRU
ISN     846           JT2=JT(I)+1
ISN     847           JB2=JB(I)-1
ISN     848           DO 7100 J=JB2,JT2
ISN     849           UN(I,J)=-UN(2*IMAX-I-1,J)
ISN     850           VN(I,J)=VN(2*IMAX-I,J)
ISN     851      7100 CONTINUE
ISN     852           IRU=IRUT
                C
                C * * START CYCLE
                C
ISN     853     11000 CONTINUE
ISN     854           FLG=1.
ISN     855           ALPHA=ULPHA
                C
                C * * COMPUTE TEMPORARY U AND V
                C
ISN     856           DO 11100 I=IRU,ILU
ISN     857           JT1=JT(I)-1
ISN     858           JB1=JB(I)
ISN     859           DO 11100 K=JB1,JT1
ISN     860           J=JT1+JB1-K
```

```
                  *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     861             IF(I .EQ. MLU)GO TO 11050
ISN     862             FUX=((UN(I,J)+UN(I+1,J))*(UN(I,J)+UN(I+1,J))+ALPHA*DABS(UN(I,J)
                       1+UN(I+1,J))*(UN(I,J)-UN(I+1,J))-(UN(I-1,J)+UN(I,J))*(UN(I-1,J)
                       2+UN(I,J))-ALPHA*DABS(UN(I-1,J)+UN(I,J))*(UN(I-1,J)-UN(I,J)))
                       3/(4.0*DELX)
ISN     863             FUY=((VN(I,J)+VN(I+1,J))*(UN(I,J)+UN(I,J+1))
                       1+ALPHA*DABS(VN(I,J)+VN(I+1,J))*(UN(I,J)-UN(I,J+1))
                       2-(VN(I,J-1)+VN(I+1,J-1))*(UN(I,J-1)+UN(I,J))
                       3-ALPHA*DABS(VN(I,J-1)+VN(I+1,J-1))*(UN(I,J-1)-UN(I,J)))/(4.*DELY)
ISN     864             VISX=NU*((UN(I+1,J)-2.*UN(I,J)+UN(I-1,J))/DELX**2+
                       1(UN(I,J+1)-2.*UN(I,J)+UN(I,J-1))/DELY**2)
ISN     865             U(I,J)=UN(I,J)+DELT*((P(I,J)-P(I+1,J))*RDX+GX-FUX-FUY    +VISX)
ISN     866       11050 CONTINUE
ISN     867             FVX=((UN(I,J)+UN(I,J+1))*(VN(I,J)+VN(I+1,J))+ALPHA*DABS(UN(I,J)
                       1+UN(I,J+1))*(VN(I,J)-VN(I+1,J))-(UN(I-1,J)+UN(I-1,J+1))*(VN(I-1,J)
                       2+VN(I,J))-ALPHA*DABS(UN(I-1,J)+UN(I-1,J+1))*(VN(I-1,J)-VN(I,J)))
                       3/(4.0*DELX)
ISN     868             FVY=((VN(I,J)+VN(I,J+1))*(VN(I,J)+VN(I,J+1))+ALPHA*DABS(VN(I,J)+VN
                       1(I,J+1))*(VN(I,J)-VN(I,J+1))-(VN(I,J-1)+VN(I,J))*(VN(I,J-1)+VN(I,J
                       2))-ALPHA*DABS(VN(I,J-1)+VN(I,J))*(VN(I,J-1)-VN(I,J)))/(4.*DELY)
ISN     869             VISY=NU*((VN(I+1,J)-2.*VN(I,J)+VN(I-1,J))/DELX**2+
                       1(VN(I,J+1)-2.*VN(I,J)+VN(I,J-1))/DELY**2)
ISN     870             V(I,J)=VN(I,J)+DELT*((P(I,J)-P(I,J+1))*RDY+GY-FVX-FVY    +VISY)
ISN     871       11100 CONTINUE
                  C
                  C * * TOP RIGID BOUNDARY CONDITIONS
                  C
ISN     872       11200 CONTINUE
ISN     873             ITU=0
ISN     874       11300 ITU=ITU+1
ISN     875             UFLG=0.
ISN     876             IF(ITU .GE. 500) GO TO 11500
ISN     877             DO 11400 I=IRU,ILU
ISN     878             J=JT(I)
ISN     879             VTM=RDY*(HTU(I)-(J-2)*DELY)
ISN     880             IF(DABS(U(I-1,J)) .LE. 1.E-6) U(I-1,J)=U(I,J)
ISN     882             F=-0.25*RDX*(H(I+1)-H(I-1))*(U(I,J)+U(I-1,J))+V(I,J-1)
                       1   -VTM*DELY*RDX*(U(I,J)-U(I-1,J))
ISN     883             DFDP= -DELT*RDY*(1.0+2.0*VTM*DELY**2*RDX**2
                       1     +0.25*RDX**2*DELY*(H(I+1)-H(I-1))*(AMAX0(0,JT(I)-JT(I-1))
                       2     -AMAX0(0,JT(I)-JT(I+1))))
ISN     884             DELP=-F/DFDP
ISN     885             IF(P(I,J) .EQ. 0.) P(I,J)=P(I,J-1)
ISN     887             CORP=DELP/P(I,J)
ISN     888             IF(DABS(CORP) .GE. 0.05 .AND. DABS(DELP) .GE. 200.) UFLG=1.
ISN     890             P(I,J)=P(I,J)+DELP
ISN     891             V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN     892             V(I,J)=V(I,J)+DELT*RDY*DELP
ISN     893             IF(I .EQ. MLU) GO TO 11350
ISN     894             U(I,J)=U(I,J)+DELT*RDX*DELP
ISN     895       11350 CONTINUE
ISN     896             IF(I .EQ. IRU) GO TO 11400
ISN     897             U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN     898       11400 CONTINUE
ISN     899             IF(UFLG .EQ. 0.) GO TO 11500
ISN     900             GO TO 11300
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     901     11500 CONTINUE
ISN     902           IF(ITU .GE. 500) GO TO 11510
ISN     903           GO TO 11520
ISN     904     11510 PRINT 25
ISN     905     11520 CONTINUE
                C
                C
ISN     906           DO 11600 I=IRU,ILU
ISN     907           JT1=JT(I)
ISN     908           JT2=JT1+1
ISN     909           U(I,JT2)=U(I,JT1)
ISN     910           V(I,JT2)=V(I,JT1)
ISN     911     11600 CONTINUE
                C
                C * * BOTTOM RIGID BOUNDARY
                C
ISN     912           ITB=0
ISN     913     11650 CONTINUE
ISN     914           BFLG=0.
ISN     915           ITB=ITB+1
ISN     916           IF(ITB .GE. 200) GO TO 11750
ISN     917           DO 11700 I=IRU,ILU
ISN     918           IF(I .EQ. MLU .OR. CODBB(I) .EQ. 0) GO TO 11700
ISN     919           J=JB(I)
ISN     920           VTM=RDY*(HB(I)-(J-2)*DELY)
ISN     921           VBM=RDY*((J-1)*DELY-HB(I))
ISN     922           F=-0.25*RDX*(HB(I+1)-HB(I-1))*(U(I,J)+U(I-1,J))+V(I,J)*VTM
                     1+VBM*(V(I,J)+DELY*RDX*(U(I,J)-U(I-1,J)))
ISN     923           DFDP=DELT*RDY*(1.0+2.0*VBM*DELY**2*RDX**2)
ISN     924           DELP=-F/DFDP
ISN     925           CORP=DELP/P(I,J)
ISN     926           IF(DABS(CORP) .GE. .05) BFLG=1.
ISN     928           P(I,J)=P(I,J)+DELP
ISN     929           U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN     930           V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN     931           V(I,J)=V(I,J)+DELT*RDY*DELP
ISN     932           U(I,J)=U(I,J)+DELT*RDX*DELP
ISN     933     11700 CONTINUE
ISN     934           IF(BFLG .EQ. 0.) GO TO 11750
ISN     935           GO TO 11650
ISN     936     11750 CONTINUE
                C
                C * * HAS CONVERGENCE BEEN REACHED
                C
ISN     937           DO 13800 ITER=1,500
                C
                C * * SET BOUNDARY CONDITIONS
                C
ISN     938           JB2=JB(ILU) -1
ISN     939           JT2=JT(ILU) +1
ISN     940           IF(ILU .LT. MLU) GO TO 11850
ISN     941           DO 11800 J=JB2,JT2
ISN     942           V(ILU+1,J)=2.*V(ILU,J)-V(ILU-1,J)
ISN     943     11800 CONTINUE
ISN     944           GO TO 11900
ISN     945     11850 CONTINUE
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     946           DO 11900 J=JB2,JT2
ISN     947           V(ILU+1,J)=V(ILU,J)
ISN     948           IF(ILU+1 .EQ. MLU)  GO TO 11900
ISN     949           U(ILU+1,J)=U(ILU,J)
ISN     950     11900 CONTINUE
ISN     951           JT1=JT(ILU )
ISN     952           U(ILU+1,JT1)=TU(ILU+1)
ISN     953           V(ILU+1,JT1)=TV(ILU+1)
                C
                C * * FREE SURFACE AND SLOPED BOUNDARY CONDITIONS
                C
ISN     954     12000 CONTINUE
ISN     955           DO 12630 I=IRU,ILU
ISN     956           IF(HB(I) .LE. HBU(I)) GO TO 12630
ISN     957           JB1=JB(I)
ISN     958           V(I,JB1-1)=V(I,JB1)+DELY*RDX*(U(I,JB1)-U(I-1,JB1))
ISN     959           U(I,JB1-1)=U(I,JB1)
ISN     960     12630 CONTINUE
                C
                C * * COMPUTE UPDATED CELL PRESSURE AND VELOCITY
                C
ISN     961           FLG=0.
ISN     962           DO 13600 I=IRU,ILU
ISN     963           JB1=JB(I)
ISN     964           JB11=JB(I)+1
ISN     965           JT11=JT(I)-1
ISN     966           DO 13600 K=JB1,JT11
ISN     967           J=JB1+JT11-K
ISN     968           IF(J .NE. JB1) GO TO 13200
ISN     969           PETA=DELY/((DFLOAT(JB1)-0.5)*DELY-HB(I))
ISN     970           DELP=(1.-PETA)*P(I,JB1+1)-P(I,JB1)
ISN     971           GO TO 13300
ISN     972     13200 CONTINUE
ISN     973           D=RDX*(U(I,J)-U(I-1,J))+RDY*(V(I,J)-V(I,J-1))
ISN     974           IF(DABS(D)      .GE.UPSI)FLG=1.0
ISN     976           DELP=-UBETA*D
ISN     977     13300 P(I,J)=P(I,J)+DELP
ISN     978           V(I,J)=V(I,J)+DELT*RDY*DELP
ISN     979           IF(I .EQ. IRU) GO TO 13400
ISN     980           U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN     981     13400 CONTINUE
ISN     982           IF(I .EQ. MLU) GO TO 13500
ISN     983           U(I,J)=U(I,J)+DELT*RDX*DELP
ISN     984     13500 CONTINUE
ISN     985           V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN     986     13600 CONTINUE
ISN     987           IF (FLG .EQ. 0.) GO TO 14000
ISN     988     13800 CONTINUE
                C
                C * * NEW POSITION OF BOTTOM SURFACE
                C
ISN     989     14000 CONTINUE
ISN     990           DO 14200 I=IRU,ILU
ISN     991           IF(I .EQ. MLU) GO TO 14210
ISN     992           JB1=JB(I)
ISN     993           HBV=RDY*(HBN(I)-FLOAT(JB1-2)*DELY)
```

```
                  *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      994            UAV=.5*(U(I-1,JB1)+U(I,JB1))
ISN      995            HB(I)=HBN(I)+DELT*(HBV*V(I,JB1)+(1.-HBV)*V(I,JB1-1)-.5*RDX*(UAV
                       1      *HBN(I+1)+GAMMA*DABS(UAV)*(HBN(I)-HBN(I+1))-UAV*HBN(I-1)
                       2      -GAMMA*DABS(UAV)*(HBN(I-1)-HBN(I))))
ISN      996            IF(HB(I) .LT. HBU(I)) HB(I)=HBU(I)
ISN      998            IF(HB(I) .GT. HTU(I)) HB(I)=HTU(I)
ISN     1000            IU=IMAX*2-I
ISN     1001            IF(HB(I) .LE. H(IU)) HB(I)=H(IU)
ISN     1003            RLH(I)=H(I)-HB(I)
ISN     1004      14200 CONTINUE
ISN     1005            GO TO 14220
ISN     1006      14210 CONTINUE
ISN     1007            FVOU=0.
ISN     1008            DO 14110 I=IRU,ILU
ISN     1009            FVOU=FVOU+(H(I)-HB(I))*DELX
ISN     1010      14110 CONTINUE
ISN     1011            HUD=(FFVOL-FVOL-FVOU)*RDX
ISN     1012            ASDT=AS*DELT
ISN     1013            IF(HUD .LT. ASDT) HUD=ASDT
ISN     1015            IF(HUD .GT. 2.*DELY) HUD=2.*DELY
ISN     1017            HB(MLU)=HBN(MLU)-HUD
ISN     1018            IF(HB(MLU) .LT. HBU(MLU)) HB(MLU)=HBU(MLU)
ISN     1020      14220 CONTINUE
                  C
                  C * * CALCULATE CELL IN WHICH SURFACE IS LOCATED AND UPDATE ARRAY
                  C
ISN     1021            DO 14250 I=IRU,ILU
ISN     1022            JBN=JB(I)
ISN     1023            JBN1=JBN-1
ISN     1024            JBO(I)=JB(I)
ISN     1025            JB(I)=    HB(I)*RDY+1.E-8+2.
ISN     1026            JB1=JB(I)
ISN     1027            IF(JBN-JB1)14224,14250,14240
ISN     1028      14224 CONTINUE
ISN     1029            J1=JB1-2
ISN     1030            DO 14230 J=JBN1,J1
ISN     1031            V(I,J)=0.
ISN     1032            U(I,J)=0.
ISN     1033            P(I,J)=0.
ISN     1034      14230 CONTINUE
ISN     1035            P(I,J1+1)=0.
ISN     1036            GO TO 14250
ISN     1037      14240 CONTINUE
ISN     1038            DO 14248 JE=JB1,JBN
ISN     1039            J=JB1+JBN-JE
ISN     1040            IF(J.GE.JT(I)) GO TO 14250
ISN     1041            TTM(I)=1.
ISN     1042            IF(I .EQ. MLU) GO TO 14243
ISN     1043            U(I,J)=(U(I+1,J)+U(I-1,J)+U(I,J+1)+U(I,J-1))*0.25
ISN     1044      14243 CONTINUE
ISN     1045            V(I,J)=(V(I,J-1)+V(I,J+1)+V(I-1,J)+V(I+1,J))*0.25
ISN     1046            P(I,J)=GY*((DFLOAT(J)-1.5)*DELY-HB(I))
ISN     1047            IF(SIGNL .LE. 450) GO TO 14248
ISN     1048            PRINT 14245,I,J,U(I,J),V(I,J)
ISN     1049      14245 FORMAT(2X,'ADDED',2(I3),3X,2(2X,1PE12.5))
ISN     1050      14248 CONTINUE
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     1051    14250 CONTINUE
ISN     1052          DO 14270 I=IRU,ILU
ISN     1053          CODBB(I)=0
ISN     1054          IF(HB(I) .NE. HBU(I)) GO TO 14270
ISN     1055          CODBB(I)=1
ISN     1056          IF(JB(I) .EQ. JBO(I)) GO TO 14270
ISN     1057          JB1=JB(I)
ISN     1058          U(I,JB1)=-BU(2 *IMAX-I)
ISN     1059          V(I,JB1)=BV(2 *IMAX-I)
ISN     1060    14270 CONTINUE
ISN     1061    14271 CONTINUE
ISN     1062          NIL=JT(ILU)-JB(ILU)
ISN     1063          IF(NIL .GE. 1) GO TO 14273
ISN     1064          ILU=ILU-1
ISN     1065          GO TO 14271
ISN     1066    14273 CONTINUE
ISN     1067          DO 14275 I=IRU,ILU
ISN     1068          IF(JT(I) .GT. JB(I)) GO TO 14275
ISN     1069          TTM(I)=1.
ISN     1070          JB(I)=JT(I)-1
ISN     1071          J=JB(I)
ISN     1072          HB(I)=(DFLOAT(J)-1.)*DELY
ISN     1073          V(I,J)=V(I,J+1)+(U(I,J+1)-U(I-1,J+1))*DELY*RDX
ISN     1074          U(I,J)=0.5*(U(I-1,J)+U(I,J+1))
ISN     1075          P(I,J)=GY*((DFLOAT(J)-1.5)*DELY-HB(I))
ISN     1076          IF(SIGNL .LE. 450) GO TO 14275
ISN     1077          PRINT 14274,I,J
ISN     1078    14274 FORMAT(10X,'MAKEUP 2ND LAYER AT I,J=',2(2X,I4))
ISN     1079    14275 CONTINUE
ISN     1080          FVOU=0.
ISN     1081          DO 14290 I=IRU,ILU
ISN     1082          FVOU=FVOU+(H(I)-HB(I))*DELX
ISN     1083    14290 CONTINUE
ISN     1084          IF(FVOU .LT. 0.) FVOU=0.
                C
                C * * NEW IL LOCATION
                C
ISN     1086    14300 CONTINUE
ISN     1087          IF(ILU .EQ. MLU) GO TO 14500
ISN     1088          IF(ILU .LT. 117) GO TO 14305
ISN     1089          JB(ILU+1)=JT(ILU+1)-1
ISN     1090          HB(ILU+1)=H(ILU+1)-1.11*DELY
ISN     1091          GO TO 14307
ISN     1092    14305 CONTINUE
ISN     1093          HB(ILU+1)=H(ILU+1)-(FFVOL-FVOL-FVOU)*RDX
ISN     1094          JB(ILU+1)=    HB(ILU+1)*RDY+2.
ISN     1095          IF(JB(ILU+1)+1 .GT. JT(ILU+1)) GO TO 14500
ISN     1096    14307 CONTINUE
ISN     1097          ILU=ILU+1
ISN     1098          JB1=JB(ILU)
ISN     1099          JT1=JT(ILU)
ISN     1100          IF(ILU .LT. 90)  GO TO 14315
ISN     1101    14310 CONTINUE
ISN     1102          DO 14312 JE=JB1,JT1
ISN     1103          J=JB1+JT1-JE
ISN     1104          U(ILU,J)=U(ILU-1,J)
```

```
              *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN    1105         V(ILU,J)=V(ILU-1,J)
ISN    1106         P(ILU,J)=GY*((DFLOAT(J)-1.5)*DELY-HB(ILU))
ISN    1107         IF(U(ILU,J) .EQ. 0. .OR. V(ILU,J) .EQ. 0.) GO TO 14322
ISN    1108   14312 CONTINUE
ISN    1109         GO TO 14324
ISN    1110   14315 CONTINUE
ISN    1111         DO 14320 JE=JB1,JT1
ISN    1112         J=JB1+JT1-JE
ISN    1113         J1=J+JT(ILU-1)-JT(ILU)
ISN    1114         J2=J+JT(ILU-2)-JT(ILU)
ISN    1115         U(ILU,J)=2.*U(ILU-1,J1 )-U(ILU-2,J2 )
ISN    1116         V(ILU,J)=2.*V(ILU-1,J1 )-V(ILU-2,J2 )
ISN    1117         P(ILU,J)=GY*((DFLOAT(J)-1.5)*DELY-HB(ILU))
ISN    1118         IF(U(ILU,J) .EQ. 0. .OR. V(ILU,J) .EQ. 0.) GO TO 14322
ISN    1119   14320 CONTINUE
ISN    1120         GO TO 14324
ISN    1121   14322 JB(ILU)=J+1
ISN    1122         HB(ILU)=(DFLOAT(J)-1.)*DELY
ISN    1123         PRINT 27
ISN    1124   14324 CONTINUE
ISN    1125         U(ILU,JT1)=TU(ILU)
ISN    1126         V(ILU,JT1)=TV(ILU)
ISN    1127         IF(ILU .EQ. MLU) GO TO 14325
ISN    1128         GO TO  14340
ISN    1129   14325 CONTINUE
ISN    1130         JT11=JT(ILU)
ISN    1131         DO 14330 J=JB1,JT11
ISN    1132         U(ILU,J)=0.
ISN    1133.        V(ILU,J)=-AS
ISN    1134   14330 CONTINUE
ISN    1135   14340 CONTINUE
ISN    1136         RLH(ILU)=H(ILU)-HB(ILU)
ISN    1137   14400 CONTINUE
ISN    1138         ILU1=ILU+1
ISN    1139         JT1=JTU(ILU1)
ISN    1140         U(ILU1,JT1)=TU(ILU1)
ISN    1141         V(ILU1,JT1)=TV(ILU1)
ISN    1142         P(ILU1,JT1)=P(ILU,JT1)
ISN    1143   14500 CONTINUE
              C
              C * * TOP RGDID BOUNDARY CONDITIONS
              C
ISN    1144         ITU=0
ISN    1145   14600 ITU=ITU+1
ISN    1146         UFLG=0.
ISN    1147         IF(ITU .GE. 200) GO TO 14750
ISN    1148         DO 14700 I=IRU,ILU
ISN    1149         J=JT(I)
ISN    1150         VTM=RDY*(HTU(I)-(J-2)*DELY)
ISN    1151         IF(DABS(U(I-1,J)) .LE. 1.E-6) U(I-1,J)=U(I,J)
ISN    1153         F=-0.25*RDX*(H(I+1)-H(I-1))*(U(I,J)+U(I-1,J))+V(I,J-1)
                   1   -VTM*DELY*RDX*(U(I,J)-U(I-1,J))
ISN    1154         DFDP= -DELT*RDY*(1.0+2.0*VTM*DELY**2*RDX**2
                   1 .   +0.25*RDX**2*DELY*(H(I+1)-H(I-1))*(AMAX0(0,JT(I)-JT(I-1))
                   2     -AMAX0(0,JT(I)-JT(I+1))))
ISN    1155         DELP=-F/DFDP
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     1156          IF(P(I,J) .EQ. 0.) P(I,J)=P(I,J-1)
ISN     1158          CORP=DELP/P(I,J)
ISN     1159          IF(DABS(CORP) .GE. 0.05 .AND. DABS(DELP) .GE. 200.) UFLG=1.
ISN     1161          P(I,J)=P(I,J)+DELP
ISN     1162          V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN     1163          V(I,J)=V(I,J)+DELT*RDY*DELP
ISN     1164          IF(I .EQ. MLU) GO TO 14650
ISN     1165          U(I,J)=U(I,J)+DELT*RDX*DELP
ISN     1166    14650 CONTINUE
ISN     1167          IF(I .EQ. IRU) GO TO 14700
ISN     1168          U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN     1169    14700 CONTINUE
ISN     1170          IF(UFLG .EQ. 0.) GO TO 14750
ISN     1171          GO TO 14600
ISN     1172    14750 CONTINUE
                C
                C * * BOTTOM RIGID BOUNDARY
                C
ISN     1173          ITB=0
ISN     1174    14780 CONTINUE
ISN     1175          BFLG=0.
ISN     1176          ITB=ITB+1
ISN     1177          IF(ITB .GE. 200) GO TO 14820
ISN     1178          DO 14800 I=IRU,ILU
ISN     1179          IF(I .EQ. MLU .OR. CODBB(I) .EQ. 0) GO TO 14800
ISN     1180          J=JB(I)
ISN     1181          VTM=RDY*(HB(I)-(J-2)*DELY)
ISN     1182          VBM=RDY*((J-1)*DELY-HB(I))
ISN     1183          F=-0.25*RDX*(HB(I+1)-HB(I-1))*(U(I,J)+U(I-1,J))+V(I,J)*VTM
                     1+VBM*(V(I,J)+DELY*RDX*(U(I,J)-U(I-1,J)))
ISN     1184          DFDP=DELT*RDY*(1.0+2.0*VBM*DELY**2*RDX**2)
ISN     1185          DELP=-F/DFDP
ISN     1186          CORP=DELP/P(I,J)
ISN     1187          IF(DABS(CORP) .GE. .05) BFLG=1.
ISN     1189          P(I,J)=P(I,J)+DELP
ISN     1190          U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN     1191          V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN     1192          V(I,J)=V(I,J)+DELT*RDY*DELP
ISN     1193          U(I,J)=U(I,J)+DELT*RDX*DELP
ISN     1194    14800 CONTINUE
ISN     1195          IF(BFLG .EQ. 0.) GO TO 14820
ISN     1196          GO TO 14780
ISN     1197    14820 CONTINUE
                C
                C
ISN     1198          DO 14850 I=IRU,ILU
ISN     1199          JT1=JT(I)
ISN     1200          JT2=JT1+1
ISN     1201          U(I,JT2)=U(I,JT1)
ISN     1202          V(I,JT2)=V(I,JT1)
ISN     1203    14850 CONTINUE
                C
                C
ISN     1204          JB2=JB(ILU+1) -1
ISN     1205          JT2=JT(ILU+1) +1
ISN     1206          IF(ILU+1-MLU) 14885,14865,14855
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      1207   14855 CONTINUE
ISN      1208         DO 14860 J=JB2,JT2
ISN      1209         V(ILU+1,J)=2.*V(ILU,J)-V(ILU-1,J)
ISN      1210   14860 CONTINUE
ISN      1211         GO TO 14900
ISN      1212   14865 CONTINUE
ISN      1213         DO 14870 J=JB2,JT2
ISN      1214         V(ILU+1,J)=V(ILU,J)
ISN      1215   14870 CONTINUE
ISN      1216         GO TO 14900
ISN      1217   14885 CONTINUE
ISN      1218         DO 14900 J=JB2,JT2
ISN      1219         U(ILU+1,J)=U(ILU,J)
ISN      1220         V(ILU+1,J)=V(ILU,J)
ISN      1221   14900 CONTINUE
ISN      1222         JT1=JT(ILU+1)
ISN      1223         U(ILU+1,JT1)=TU(ILU+1)
ISN      1224         V(ILU+1,JT1)=TV(ILU+1)
                C
                C * * FREE SURFACE AND VOLUME
                C
ISN      1225         DO 15000 I=IRU,ILU
ISN      1226         IF(HB(I) .LE. HBU(I)) GO TO 15000
ISN      1227         JB1=JB(I)
ISN      1228         IF(TTM(I) .NE. 1.) GO TO 14950
ISN      1229         V(I,JB1-1)=(V(I,JB1)+V(I,JB1-2)+V(I-1,JB1-1)+V(I+1,JB1-1))*0.25
ISN      1230         IF(I .EQ. MLU) GO TO 15000
ISN      1231         U(I,JB1-1)=(U(I,JB1)+U(I,JB1-2)+U(I-1,JB1-1)+U(I+1,JB1-1))*0.25
ISN      1232         GO TO 15000
ISN      1233   14950 CONTINUE
ISN      1234         V(I,JB1-1)=V(I,JB1)+DELY*RDX*(U(I,JB1)-U(I-1,JB1))
ISN      1235         U(I,JB1-1)=(U(I,JB1)+U(I,JB1-2)+U(I-1,JB1-1)+U(I+1,JB1-1))*0.25
ISN      1236   15000 CONTINUE
ISN      1237         FVOU=0.
ISN      1238         DO 15100 I=IRU,ILU
ISN      1239         FVOU=FVOU+(H(I)-HB(I))*DELX
ISN      1240   15100 CONTINUE
ISN      1241         PRINT 39, ITER,T,SIGNL,FVOU
                C
                C * * TRANSFER VELOCITY TO LOWER HALF
                C
ISN      1242         DO 15200 IU=IRU,ILU
ISN      1243         I=2*IMAX-IU
ISN      1244         IF(HB(IU) .GT. H(I)) GO TO 15150
ISN      1245         H(I)= H(IU)
ISN      1246         HN(I)=H(I)
ISN      1247         JT(I)=JT(IU)
ISN      1248   15150 CONTINUE
ISN      1249         JB1=JB(IU)
ISN      1250         JT1=JT(IU)
ISN      1251         DO 15200 J=JB1,JT1
ISN      1252         U(I-1,J)=-U(IU,J)
ISN      1253         UN(I,J)=U(I,J)
ISN      1254         V(I,J)=V(IU,J)
ISN      1255         VN(I,J)=V(I,J)
ISN      1256         P(I,J)=P(IU,J)
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     1257    15200 CONTINUE
                C
                C * * CHECK REACH BOTTOM SURFACE
                C
ISN     1258          IF(ILU .LT. MLU) GO TO 15260
ISN     1259          ILL1=0
ISN     1260          DO 15250 I=IRU,ILU
ISN     1261          IF(CODBB(I) .NE. 1) GO TO 15250
ISN     1262          IF(ILL1 .NE. 0) GO TO 15240
ISN     1263          ILL1=2*IMAX-I
ISN     1264          HN(ILL1)=H(I)
ISN     1265          H(ILL1)=H(I)
ISN     1266          JT(ILL1)=JT(I)
ISN     1267          JB(ILL1)=JB(I)
ISN     1268          JB2=JB(I)-1
ISN     1269          JT2=JT(I)+1
ISN     1270          DO 15230 J=JB2,JT2
ISN     1271          U(ILL1,J)=-U(I-1,J)
ISN     1272          U(ILL1-1,J)=-U(I,J)
ISN     1273          V(ILL1,J)=V(I,J)
ISN     1274          V(ILL1-1,J)=V(I+1,J)
ISN     1275          P(ILL1,J)=P(I,J)
ISN     1276    15230 CONTINUE
ISN     1277          JB1=JB(ILL1)
ISN     1278          U(ILL1,JB1)=BU(ILL1)
ISN     1279          V(ILL1,JB1)=BV(ILL1)
ISN     1280          V(ILL1,JB1-1)=V(ILL1,JB1)+DELY*RDX*(U(ILL1,JB1)-U(ILL1-1,JB1))
ISN     1281          U(ILL1,JB1-1)=(AS-V(ILL1,JB1-1)*VSIN(ILL1))/VCOS(ILL1)
ISN     1282    15240 CONTINUE
ISN     1283          IU=2*IMAX-I
ISN     1284          CODTB(IU)=1
ISN     1285    15250 CONTINUE
ISN     1286    15260 CONTINUE
                C
                C * * CALCULATE VELOCITY VECTOR
                C
ISN     1287          IF(TTU+1.E-6 .LT. TUPRT) GO TO 16000
ISN     1288          TUPRT=TUPRT+CUPRT*DELT
ISN     1289          DO 15300 I=IRU,ILU
ISN     1290          JB2=JB(I)-1
ISN     1291          JT2=JT(I)+1
ISN     1292          DO 15300 J=JB2,JT2
ISN     1293          UV(I,J)=DSQRT(U(I,J)*U(I,J)+V(I,J)*V(I,J))
ISN     1294          IF(DABS(U(I,J)) .LE. 1.E-6) U(I,J)=1.E-6
ISN     1296          ANG(I,J)=57.3*DATAN(V(I,J)/U(I,J))
ISN     1297          IF(U(I,J) .EQ. 1.E-6) U(I,J)=0.
ISN     1299    15300 CONTINUE
                C
                C * * LIST VELOCITY, PRESSURE, AND SURFACE POSITION
                C
ISN     1300          PRINT 35
ISN     1301          PRINT 47
ISN     1302          DO 15900 I=IRU,ILU
ISN     1303          JB1=JB(I)
ISN     1304          JT1=JT(I)
ISN     1305          IB=2*IMAX-I
```

```
               *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     1306          JB2=JB(I)-1
ISN     1307          JT2=JT(I)+1
ISN     1308          DO 15900 J=JB2,JT2
ISN     1309          PRINT 38,I,IB,J,U(I,J),V(I,J),P(I,J), HB(I),JT1,JB1,UV(I,J),
                     .1ANG(I,J)
ISN     1310   15900 CONTINUE
               C
               C * * SET THE ADVANCE TIME VELOCITIES U AND V INTO THE UN AND VN ARRAYS
               C * * AND THE ADVANCE TIME SURFACE HEIGHT H INTO THE HN ARRAY
               C
ISN     1311   16000 CONTINUE
ISN     1312          IRU1=IRU-1
ISN     1313          ILU1=ILU+1
ISN     1314          DO 16100 I=IRU1,ILU1
ISN     1315          TTM(I)=0.
ISN     1316          JB2=JB(I)-1
ISN     1317          JT2=JT(I)+1
ISN     1318          DO 16100 J=JB2,JT2
ISN     1319          UN(I,J)=U(I,J)
ISN     1320          VN(I,J)=V(I,J)
ISN     1321          HBN(I)=HB(I)
ISN     1322   16100 CONTINUE
               C
               C * * ADVANCE TIME T=T+DELT
               C
ISN     1323          TTU=TTU+DELT
ISN     1324          SIGNL=SIGNL+1
ISN     1325   16200 CONTINUE
               C
               C * * INPUT TO LOWER SECTION
               C
ISN     1326          IF(ILL1 .EQ. 0) GO TO 20000
ISN     1327          IF(MEET .EQ. 0) GO TO 16300
ISN     1328          IL=ILL1+1
ISN     1329          LL=IL
ISN     1330          FLOL=0.
ISN     1331          GO TO 20000
ISN     1332   16300 CONTINUE
ISN     1333          TDVOL=FFVOL-FVOL-FVOU-FLOL
ISN     1334          ILL=ILL1+1
ISN     1335          IF(ILL .LE. LR) GO TO 17000
ISN     1336   16400 CONTINUE
ISN     1337          IF(TDVOL .LE. 0.) GO TO 20000
ISN     1338          H(ILL)=HB(ILL)+TDVOL*RDX
ISN     1339          JT(ILL)=H(ILL)*RDY+1.E-6 +2.
ISN     1340          PRINT 29, ILL1,JT(ILL),H(ILL)
ISN     1341          IF(JT(ILL) .EQ. JB(ILL)) GO TO 20000
ISN     1342          LL=ILL
ISN     1343          HN(LL)=H(LL)
ISN     1344          LR=LL
ISN     1345          JT1=JT(LL)
ISN     1346          JB1=JB(LL)
ISN     1347          IF(ILL1 .NE. ML) GO TO 16470
ISN     1348          DO 16450 J=JB1,JT1
ISN     1349          U(LL,J)=0.5*U(ML,J)
ISN     1350          V(LL,J)=0.5*V(ML,J)
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     1351          P(LL,J)=-GY*(H(LL)-(DFLOAT(J)-1.5)*DELY)
ISN     1352    16450 CONTINUE
ISN     1353          U(LL,JB1)=TU(LL)
ISN     1354          V(LL,JB1)=TV(LL)
ISN     1355          CODBB(LL)=1
ISN     1356          GO TO 16500
ISN     1357    16470 CONTINUE
ISN     1358          DO 16490 J=JB1,JT1
ISN     1359          J1=J-JB(LL)+JB(LL-1)
ISN     1360          J2=J-JB(LL)+JB(LL-2)
ISN     1361          U(LL,J)=2.*U(LL-1,J1)-U(LL-2,J2)
ISN     1362          V(LL,J)=2.*V(LL-1,J1)-V(LL-2,J2)
ISN     1363          P(LL,J)=-GY*(H(LL)-(DFLOAT(J)-1.5)*DELY)
ISN     1364    16490 CONTINUE
ISN     1365    16500 CONTINUE
ISN     1366          CODBB(LL)=1
ISN     1367          U(LL,JB1)=TU(LL)
ISN     1368          V(LL,JB1)=TV(LL)
ISN     1369          V(LL,JB1-1)=V(LL,JB1)+(U(LL,JB1)-U(LL-1,JB1))*DELY*RDX
ISN     1370          U(LL,JB1-1)=(AS-V(LL,JB1-1)*VSIN(LL))/VCOS(LL)
                C
                C * * BOTTOM BOUNDARY CONDITION
                C
ISN     1371          DO 16600 ITB=1,100
ISN     1372          BFLG=0.
ISN     1373          DO 16550 I=LL,LR
ISN     1374          J=JB(I)
ISN     1375          VTM=RDY*(HB(I)-(J-2)*DELY)
ISN     1376          VBM=RDY*((J-1)*DELY-HB(I))
ISN     1377          F=-0.25*RDX*(HB(I+1)-HB(I-1))*(U(I,J)+U(I-1,J))+V(I,J)*VTM
                     1+VBM*(V(I,J)+DELY*RDX*(U(I,J)-U(I-1,J)))
ISN     1378          DFDP=DELT*RDY*(1.0+2.0*VBM*DELY**2*RDX**2)
ISN     1379          DELP=-F/DFDP
ISN     1380          CORP=DELP/P(I,J)
ISN     1381          IF(DABS(CORP) .GE. .05 .AND. DABS(DELP) .GE. 200) BFLG=1.
ISN     1383          P(I,J)=P(I,J)+DELP
ISN     1384          V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN     1385          V(I,J)=V(I,J)+DELT*RDY*DELP
ISN     1386          U(I,J)=U(I,J)+DELT*RDX*DELP
ISN     1387          IF(I .EQ. LL) GO TO 16550
ISN     1388          U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN     1389    16550 CONTINUE
ISN     1390          IF(BFLG .EQ. 0.) GO TO 16650
ISN     1391    16600 CONTINUE
ISN     1392    16650 CONTINUE
ISN     1393          V(LL,JB1-1)=V(LL,JB1)+(U(LL,JB1)-U(LL-1,JB1))*DELY*RDX
ISN     1394          U(LL,JB1-1)=(AS-V(LL,JB1-1)*VSIN(LL))/VCOS(LL)
                C
                C * * FREE SURFACE CONDITIONS
                C
ISN     1395          DO 16700 I=LL,LR
ISN     1396          JT1=JT(I)
ISN     1397          V(I,JT1)=V(I,JT1-1)-DELY*RDX*(U(I,JT1)-U(I-1,JT1))
ISN     1398          U(I,JT1+1)=U(I,JT1)
ISN     1399    16700 CONTINUE
ISN     1400          JB2=JB(LR)-1
```

```
                 *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN    1401            JT2=JT(LR)+1
ISN    1402            LR1=LR+1
ISN    1403            DO 16800 J=JB2,JT2
ISN    1404            U(LR1,J)=U(LR,J)
ISN    1405            V(LR1,J)=V(LR,J)
ISN    1406      16800 CONTINUE
ISN    1407            JB2=JB(LR1)-1
ISN    1408            U(LR1,JB2+1)=BU(LR1)
ISN    1409            V(LR1,JB2+1)=BV(LR1)
ISN    1410            GO TO 19700
ISN    1411      17000 CONTINUE
ISN    1412            LL=ILL
ISN    1413            FLG=1.
ISN    1414            ALPHA=DLPHA
                 C
                 C * * COMPUTE TEMPORARY U AND V
                 C
ISN    1415            DO 17100 I=LL,LR
ISN    1416            JT1= JT(I)
ISN    1417            JB1=JB(I)+1
ISN    1418            DO 17100 J=JB1,JT1
ISN    1419            FUX=((UN(I,J)+UN(I+1,J))*(UN(I,J)+UN(I+1,J))+ALPHA*DABS(UN(I,J)
                      1+UN(I+1,J))*(UN(I,J)-UN(I+1,J))-(UN(I-1,J)+UN(I,J))*(UN(I-1,J)
                      2+UN(I,J))-ALPHA*DABS(UN(I-1,J)+UN(I,J))*(UN(I-1,J)-UN(I,J)))
                      3/(4.0*DELX)
ISN    1420            FUY=((VN(I,J)+VN(I+1,J))*(UN(I,J)+UN(I,J+1))
                      1+ALPHA*DABS(VN(I,J)+VN(I+1,J))*(UN(I,J)-UN(I,J+1))
                      2-(VN(I,J-1)+VN(I+1,J-1))*(UN(I,J-1)+UN(I,J))
                      3-ALPHA*DABS(VN(I,J-1)+VN(I+1,J-1))*(UN(I,J-1)-UN(I,J)))/(4.*DELY)
ISN    1421            VISX=NU*((UN(I+1,J)-2.*UN(I,J)+UN(I-1,J))/DELX**2+
                      1(UN(I,J+1)-2.*UN(I,J)+UN(I,J-1))/DELY**2)
ISN    1422            U(I,J)=UN(I,J)+DELT*((P(I,J)-P(I+1,J))*RDX+GX-FUX-FUY    +VISX)
ISN    1423            FVX=((UN(I,J)+UN(I,J+1))*(VN(I,J)+VN(I+1,J))+ALPHA*DABS(UN(I,J)
                      1+UN(I,J+1))*(VN(I,J)-VN(I+1,J))-(UN(I-1,J)+UN(I-1,J+1))*(VN(I-1,J)
                      2+VN(I,J))-ALPHA*DABS(UN(I-1,J)+UN(I-1,J+1))*(VN(I-1,J)-VN(I,J)))
                      3/(4.0*DELX)
ISN    1424            FVY=((VN(I,J)+VN(I,J+1))*(VN(I,J)+VN(I,J+1))+ALPHA*DABS(VN(I,J)+VN
                      1(I,J+1))*(VN(I,J)-VN(I,J+1))-(VN(I,J-1)+VN(I,J))*(VN(I,J-1)+VN(I,J
                      2))-ALPHA*DABS(VN(I,J-1)+VN(I,J))*(VN(I,J-1)-VN(I,J)))/(4.*DELY)
ISN    1425            VISY=NU*((VN(I+1,J)-2.*VN(I,J)+VN(I-1,J))/DELX**2+
                      1(VN(I,J+1)-2.*VN(I,J)+VN(I,J-1))/DELY**2)
ISN    1426            V(I,J)=VN(I,J)+DELT*((P(I,J)-P(I,J+1))*RDY+GY-FVX-FVY    +VISY)
ISN    1427      17100 CONTINUE
                 C
                 C *.* BOTTOM BOUNDARY CONDITIONS
                 C
ISN    1428            DO 17200 ITB=1,100
ISN    1429            BFLG=0.
ISN    1430            DO 17150 I=LL,LR
ISN    1431            J=JB(I)
ISN    1432            VTM=RDY*(HB(I)-(J-2)*DELY)
ISN    1433            VBM=RDY*((J-1)*DELY-HB(I))
ISN    1434            F=-0.25*RDX*(HB(I+1)-HB(I-1))*(U(I,J)+U(I-1,J))+V(I,J)*VTM
                      1+VBM*(V(I,J)+DELY*RDX*(U(I,J)-U(I-1,J)))
ISN    1435            DFDP=DELT*RDY*(1.0+2.0*VBM*DELY**2*RDX**2)
ISN    1436            DELP=-F/DFDP
```

```
                 *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      1437          CORP=DELP/P(I,J)
ISN      1438          IF(DABS(CORP) .GE. .05 .AND. DABS(DELP) .GE. 200) BFLG=1.
ISN      1440          P(I,J)=P(I,J)+DELP
ISN      1441          V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN      1442          V(I,J)=V(I,J)+DELT*RDY*DELP
ISN      1443          U(I,J)=U(I,J)+DELT*RDX*DELP
ISN      1444          IF(I .EQ. LL) GO TO 17150
ISN      1445          U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN      1446    17150 CONTINUE
ISN      1447          IF(BFLG .EQ. 0.) GO TO 17300
ISN      1448    17200 CONTINUE
                 C
                 C * * HAS CONVERGENCE BEEN REACHED
                 C
ISN      1449    17300 CONTINUE
ISN      1450          DO 18000 ITER=1,200
                 C
                 C * * SET LEFT AND RIGHT BOUNDARY CONDITIONS
                 C
ISN      1451          JB2=JB(LR)-1
ISN      1452          JT2=JT(LR)+1
ISN      1453          DO 17400 J=JB2,JT2
ISN      1454          U(LR+1,J)=U(LR,J)
ISN      1455          V(LR+1,J)=V(LR,J)
ISN      1456    17400 CONTINUE
ISN      1457          JB1=JB(LR+1)
ISN      1458          U(LR+1,JB1)=BU(LR+1)
ISN      1459          V(LR+1,JB1)=BV(LR+1)
                 C
                 C * * FREE SURFACE CONDITIONS
                 C
ISN      1460          DO 17500 I=LL,LR
ISN      1461          JT1=JT(I)
ISN      1462          IF(I .LE. 6) GO TO 17450
ISN      1463          IF(JT(I+1) .LT. JT(I)) U(I,JT1)=U(I,JT1-1)
ISN      1465    17450 CONTINUE
ISN      1466          V(I,JT1)=V(I,JT1-1)-DELY*RDX*(U(I,JT1)-U(I-1,JT1))
ISN      1467          U(I,JT1+1)=U(I,JT1)
ISN      1468    17500 CONTINUE
                 C
                 C * * COMPUTE UPDATED CELL PRESSURE AND VELOCITY
                 C
ISN      1469          FLG=0.
ISN      1470          DO 17800 I=LL,LR
ISN      1471          JT1=JT(I)
ISN      1472          JB1=JB(I)+1
ISN      1473          DO 17800 J=JB1,JT1
ISN      1474          IF(J .NE. JT1) GO TO 17600
ISN      1475          PETA=DELY/(HN(I)-(DFLOAT(JT1)-2.5)*DELY)
ISN      1476          DELP=(1.0-PETA)*P(I,JT1-1)-P(I,JT1)
ISN      1477          GO TO 17700
ISN      1478    17600 CONTINUE
ISN      1479          D=RDX*(U(I,J)-U(I-1,J))+RDY*(V(I,J)-V(I,J-1))
ISN      1480          IF(DABS(D)     .GE.EPSI)FLG=1.0
ISN      1482          DELP=-BETA*D
ISN      1483    17700 P(I;J)=P(I,J)+DELP
```

```
                 *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     1484           U(I,J)=U(I,J)+DELT*RDX*DELP
ISN     1485           IF(I .EQ. LL) GO TO 17750
ISN     1486           U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN     1487     17750 CONTINUE
ISN     1488           V(I,J)=V(I,J)+DELT*RDY*DELP
ISN     1489           IF(J .EQ. JB1) GO TO 17800
ISN     1490           V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN     1491     17800 CONTINUE
ISN     1492           IF(FLG .EQ. 0.) GO TO 18100
ISN     1493     18000 CONTINUE
                 C
                 C * * COMPUTE NEW SURFACE POSITION
                 C
ISN     1494     18100 CONTINUE
ISN     1495           DO 18200 I=LL,LR
ISN     1496           JT1=JT(I)
ISN     1497           HV=RDY*(HN(I)-DFLOAT(JT1-2)*DELY)
ISN     1498           UAV=0.5*(U(I-1,JT1) +U(I,JT1))
ISN     1499           H(I)=HN(I)+DELT*(HV*V(I,JT1)+(1.0-HV)*V(I,JT1-1)
                      1     -0.5*RDX*(UAV*HN(I+1)+GAMMA*DABS(UAV)*(HN(I)-HN(I+1))
                      2     -UAV*HN(I-1)-GAMMA*DABS(UAV)*(HN(I-1)-HN(I))))
ISN     1500           IF(H(I) .LT. HB(I)) H(I)=HB(I)
ISN     1502     18200 CONTINUE
                 C
                 C * * CALCULATE CELL IN WHICH SURFACE IS LOCATED AND UPDATE ARRAY
                 C
ISN     1503           DO 18500 I=LL,LR
ISN     1504           JTO(I)=JT(I)
ISN     1505           JT(I)=    H(I)*RDY+1.0E-8+2.
ISN     1506           IF(JT(I) .EQ. JTO(I)) GO TO 18500
ISN     1507           JT1=JT(I)
ISN     1508           IF(JT1 .GT. JTO(I)) GO TO 18400
ISN     1509           J1=JT1+1
ISN     1510           JTO1=JTO(I)
ISN     1511           DO 18300 J=J1,JTO1
ISN     1512           U(I,J)=0.
ISN     1513           V(I,J)=0.
ISN     1514           P(I,J)=0.
ISN     1515     18300 CONTINUE
ISN     1516           GO TO 18500
ISN     1517     18400 CONTINUE
ISN     1518           JT11=JTO(I)+1
ISN     1519           DO 18500 J=JT11,JT1
ISN     1520           P(I,J)=-GY*(H(I)-(DFLOAT(J)-1.5)*DELY)
ISN     1521     18500 CONTINUE
                 C
                 C * * CALCULATE NEW VOLUME
                 C
ISN     1522           LR2=LR-1
ISN     1523           DO 18600 I=LL,LR2
ISN     1524           IF(JT(I) .GT. JB(I)) GO TO 18600
ISN     1525           JT(I)=JB(I)+1
ISN     1526           J=JT(I)
ISN     1527           H(I)=(DFLOAT(J)-2.)*DELY
ISN     1528           V(I,J)=0.5*(V(I-1,J)+V(I+1,J))
ISN     1529           U(I,J)=U(I-1,J)+(V(I,J-1)-V(I,J))*DELX*RDY
```

```
                  *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     1530            P(I,J)=-GY*(H(I)-(DFLOAT(J)-1.5)*DELY)
ISN     1531      18600 CONTINUE
ISN     1532            FLOL=0.
ISN     1533            DO 18700 I=LL,LR
ISN     1534            FLOL=FLOL+(H(I)-HB(I))*DELX
ISN     1535      18700 CONTINUE
                  C
                  C * * NEW IR LOCATION
                  C
ISN     1536            NIR=JT(LR)-JB(LR)
ISN     1537            IF(NIR .LE. 0) GO TO 18900
ISN     1538            IF(LR+2 .LT. IL) GO TO 18730
ISN     1539            MEET=1
ISN     1540            H(LR+1)=0.5*(H(LR)+H(IL))
ISN     1541            JT(LR+1)=H(LR+1)*RDY+2.
ISN     1542            LR=LR+1
ISN     1543            PRINT 42, LR,JT(LR),H(LR)
ISN     1544            CODBB(LR)=1
ISN     1545            JT1=JT(LR)
ISN     1546            JB1=JB(LR)
ISN     1547            DO 18710 J=JB1,JT1
ISN     1548            P(LR,J)=-GY*(H(LR)-(DFLOAT(J)-1.5)*DELY)
ISN     1549            U(LR,J)=0.5*(U(LR-1,J)+U(LR+1,J))
ISN     1550            V(LR,J)=0.5*(V(LR-1,J)+V(LR+1,J))
ISN     1551      18710 CONTINUE
ISN     1552            U(LR,JB1)=BU(LR)
ISN     1553            V(LR,JB1)=BV(LR)
ISN     1554            V(LR,JB1-1)=V(LR,JB1)+(U(LR,JB1)-U(LR-1,JB1  ))*DELY*RDX
ISN     1555            U(LR,JB1-1)=(AS-V(LR,JB1-1)*VSIN(LR))/VCOS(LR)
ISN     1556            GO TO 19100
ISN     1557      18730 CONTINUE
ISN     1558            H(LR+1)=HB(LR+1)+(FFVOL-FVOL-FVOU-FLOL)*RDX
ISN     1559            JT(LR+1)=H(LR+1)*RDY+2.
ISN     1560            IF(JT(LR+1) .LE. JB(LR+1)) GO TO 19000
ISN     1561            LR=LR+1
ISN     1562            PRINT 42, LR,JT(LR),H(LR)
ISN     1563            CODBB(LR)=1
ISN     1564            JT1=JT(LR)
ISN     1565            JB1=JB(LR)
ISN     1566            DO 18800 J=JB1,JT1
ISN     1567            P(LR,J)=-GY*(H(LR)-(DFLOAT(J)-1.5)*DELY)
ISN     1568            U(LR,J)=U(LR-1,J)
ISN     1569            V(LR,J)=V(LR-1,J)
ISN     1570      18800 CONTINUE
ISN     1571            U(LR,JB1)=BU(LR)
ISN     1572            V(LR,JB1)=BV(LR)
ISN     1573            RLH(LR)=H(LR)-HB(LR)
ISN     1574            V(LR,JB1-1)=V(LR,JB1)+(U(LR,JB1)-U(LR-1,JB1  ))*DELY*RDX
ISN     1575            U(LR,JB1-1)=(AS-V(LR,JB1-1)*VSIN(LR))/VCOS(LR)
ISN     1576            GO TO 19000
ISN     1577      18900 CODBB(LR)=0
ISN     1578            LR=LR-1
ISN     1579            IF(JT(LR)-JB(LR) .LT. 1) GO TO 18900
ISN     1580      19000 CONTINUE
ISN     1581            LR1=LR+1
ISN     1582            JB1=JB(LR1)
```

```
                   *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN      1583            U(LR1,JB1)=BU(LR1)
ISN      1584            V(LR1,JB1)=BV(LR1)
ISN      1585            P(LR1,JB1)=P(LR,JB1)
                   C
                   C * * BOTTOM BOUNDARY CONDITIONS
                   C
ISN      1586      19100 CONTINUE
ISN      1587            DO 19300 ITB=1,100
ISN      1588            BFLG=0.
ISN      1589            DO 19200 I=LL,LR
ISN      1590            J=JB(I)
ISN      1591            VTM=RDY*(HB(I)-(J-2)*DELY)
ISN      1592            VBM=RDY*((J-1)*DELY-HB(I))
ISN      1593            F=-0.25*RDX*(HB(I+1)-HB(I-1))*(U(I,J)+U(I-1,J))+V(I,J)*VTM
                        1+VBM*(V(I,J)+DELY*RDX*(U(I,J)-U(I-1,J)))
ISN      1594            DFDP=DELT*RDY*(1.0+2.0*VBM*DELY**2*RDX**2)
ISN      1595            DELP=-F/DFDP
ISN      1596            CORP=DELP/P(I,J)
ISN      1597            IF(DABS(CORP) .GE. .05 .AND. DABS(DELP) .GE. 200) BFLG=1.
ISN      1599            P(I,J)=P(I,J)+DELP
ISN      1600            V(I,J-1)=V(I,J-1)-DELT*RDY*DELP
ISN      1601            V(I,J)=V(I,J)+DELT*RDY*DELP
ISN      1602            U(I,J)=U(I,J)+DELT*RDX*DELP
ISN      1603            IF(I .EQ. LL) GO TO 19200
ISN      1604            U(I-1,J)=U(I-1,J)-DELT*RDX*DELP
ISN      1605      19200 CONTINUE
ISN      1606            IF(BFLG .EQ. 0.) GO TO 19400
ISN      1607      19300 CONTINUE
                   C
                   C * * IR+1 VELOCITY
                   C
ISN      1608      19400 CONTINUE
ISN      1609            IF(LR+1 .EQ. IL) GO TO 19550
ISN      1610            JB2=JB(LR)-1
ISN      1611            JT2=JT(LR)+1
ISN      1612            LR1=LR+1
ISN      1613            DO 19500 J=JB2,JT2
ISN      1614            U(LR1,J)=U(LR,J)
ISN      1615            V(LR1,J)=V(LR,J)
ISN      1616      19500 CONTINUE
ISN      1617            U(LR1,JB2+1)=BU(LR1)
ISN      1618            V(LR1,JB2+1)=BV(LR1)
                   C
                   C * * FREE SURFACE CONDITION
                   C
ISN      1619      19550 CONTINUE
ISN      1620            DO 19600 I=LL,LR
ISN      1621            JT1=JT(I)
ISN      1622            IF(I .LE. 6) GO TO 19570
ISN      1623            IF(JT(I+1) .LT. JT(I)) U(I,JT1)=U(I,JT1-1)
ISN      1625      19570 CONTINUE
ISN      1626            V(I,JT1)=V(I,JT1-1)-DELY*RDX*(U(I,JT1)-U(I-1,JT1))
ISN      1627            U(I,JT1+1)=U(I,JT1)
ISN      1628      19600 CONTINUE
                   C
                   C * * CALCULATE TOTAL VOLUME
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

                C
ISN     1629    19700 CONTINUE
ISN     1630          FLOL=0.
ISN     1631          DO 19800 I=LL,LR
ISN     1632          CODBB(I)=1
ISN     1633          FLOL=FLOL+(H(I)-HB(I))*DELX
ISN     1634    19800 CONTINUE
ISN     1635          PRINT49,ITER,TTD,LYCLE,FLOL
                C
                C * * TRANSFER TO UPPER SECTION
                C
ISN     1636          I=LL
ISN     1637          IU=2*IMAX-I
ISN     1638          JB2=JB(I)-1
ISN     1639          JT2=JT(I)+1
ISN     1640          DO 19850 J=JB2,JT2
ISN     1641          U(IU,J)=-U(I-1,J)
ISN     1642          V(IU,J)=V(I,J)
ISN     1643    19850 CONTINUE
                C
                C * * CALCULATE VELOCITY VECTOR
                C
ISN     1644          IF(TTD+1.E-6 .LT. TLPRT) GO TO 19930
ISN     1645          TLPRT=TLPRT+CLPRT*DELT
ISN     1646          DO 19900 I=ILL1,LR
ISN     1647          JB2=JB(I)-1
ISN     1648          JT2=JT(I)+1
ISN     1649          DO 19900 J=JB2,JT2
ISN     1650          UV(I,J)=DSQRT(U(I,J)*U(I,J)+V(I,J)*V(I,J))
ISN     1651          IF(DABS(U(I,J)) .LE. 1.E-6) U(I,J)=1.E-6
ISN     1653          ANG(I,J)=57.3*DATAN(V(I,J)/U(I,J))
ISN     1654          IF(U(I,J) .EQ. 1.E-6) U(I,J)=0.
ISN     1656    19900 CONTINUE
                C
                C * * LIST VELOCITY, PRESSURE, AND SURFACE POSITION
                C
ISN     1657          PRINT 35
ISN     1658          PRINT 47
ISN     1659          DO 19920 I=ILL1,LR
ISN     1660          JT1=JT(I)
ISN     1661          JT2=JT1+1
ISN     1662          JB1=JB(I)
ISN     1663          JB2=JB1-1
ISN     1664          DO 19920 J=JB2,JT2
ISN     1665          PRINT 48,I,J,U(I,J),V(I,J),P(I,J),H(I),JT1,JB1,UV(I,J),ANG(I,J)
ISN     1666    19920 CONTINUE
                C
                C * * SET THE ADVANCE TIME VELOCITIES U AND V INTO THE UN AND VN ARRAYS
                C * * AND THE ADVANCE TIME SURFACE HEIGHT H INTO THE HN ARRAY
                C
ISN     1667    19930 CONTINUE
ISN     1668          LR1=LR+1
ISN     1669          DO 19940 I=ILL1,LR1
ISN     1670          JT2=JT(I)+1
ISN     1671          JB2=JB(I)-1
ISN     1672          DO 19940 J=JB2,JT2
```

```
                *....*...1.........2.........3.........4.........5.........6.........7.*.......8

ISN     1673          UN(I,J)=U(I,J)
ISN     1674          VN(I,J)=V(I,J)
ISN     1675          HN(I)=H(I)
ISN     1676    19940 CONTINUE
                C
                C * * ADVANCE TIME T=T+DELT
                C
ISN     1677          TTD=TTD+DELT
ISN     1678          LYCLE=LYCLE+1
ISN     1679    20000 CONTINUE
ISN     1680          IF(ITER .GT. 500) GO TO 20500
ISN     1681          T=T+DELT
ISN     1682    20300 CONTINUE
ISN     1683    20500 STOP
ISN     1684          END

*STATISTICS*    SOURCE STATEMENTS = 1638, PROGRAM SIZE = 381744 BYTES, PROGRAM NAME = MAIN    PAGE:    1.

*STATISTICS*     NO DIAGNOSTICS GENERATED.

****** END OF COMPILATION 1 ******
```

BIBLIOGRAPHY

1. Bjorksten Research Laboratories, Polyesters and Their Applications. New York: Reinhold Publishing Corp., 1956.

2. Parkyn.B., Lamb, F., and Clifton, B.V., Polyester, Vol.2, Unsaturated Polyesters and Ployester Plasticisers. New York: American Elsevier Publishing Co., Inc. 1967.

3. Ramazzotti, D.J., "Rotational Molding," Modern Plastics Encyclopedia, 1978-1979, 372.

4. Rao, M.A and Throne, J.L., Polym. Eng. Sci., 12(1972), 237-264.

5. Progelhof, R.C. and Throne, J.L., Polym. Eng. Sci., 16(1976) 680-686.

6. __________, paper presented at First Internations Conference on Polymer Processing, MIT, August, 1977.

7. Throne, J.L., Gianchandani, J., and Progelhof, R.C., Proc. Second World Congress of Chemical Engineering, Vol. VI, Montreal, October 4 - 9, 1981, 465-471.

8. Throne, J.L., and Gianchandani, J., Polym. Eng. Sci., 20(1980) 899-919.

9. Progelhof, R.C. and Throne, J.L., Polym. Eng. Sci., 15(1975) 690-695.

10. Throne. L.J., Polym. Eng. Sci., 16(1976) 257-264.

11. _______, Polym. Eng. Sci., 12(1972) 335-339.

12. Broyer, E. and Macosko, C.W., AIChE, J., 22(1976) 268-276.

13. White, R.E., Tappi 39(1956) 228.

14. White, R.E., and Higgins, T.W., Tappi, 41(1958) 71.

15. Malkin, B.A., Dominion Eng., March (1937)

16. Balmer, R.T., Nature, 227(1970) 600-601.

17. Karweit, M.J. and Corrsin, J., Phys. Fluids, 18(1975) 111-112.

18. Balmer, R.T. and Wang, P.G., J. Fluid Eng., 98(1976) 688-694.

19. Phillips, O.M., J. Fluid Mech., 7(1960) 340-352.

20. Cerro, R.L. and Scriven, L.E., AIChE Regional Meeting, Bloomington, Minn., Feb. 1971.

21. Ruschak, K.J. and Scriven, L.E., J. Fluid Mech., 76(1976) 113-125.

BIBLIOGRAPHY Cont'd

22. Greenspan, H.P., J. Fluid Mech., 74(1976) 335-351.

23. Gans, R.F., J. Fluid Mech., 82(1977) 415-427.

24. Deiber, J.A. and Cerro, R.L., Ind. Eng. Chem., Fundam. 15(1976) 102-110.

25. Orr, F.M., and Scriven L.E., J. Fluid Mech., 84(1978) 145-165.

26. Garabedian, P.R., Partial Differential Equations. New York: John Wiley & Sons, Inc., 1964.

27. Harlow, F.H. and Amsden A.A., J. Comput. Phys. 8(1971) 197-213.

28. Hirt, C.W., Nichols, B.D., and Romero, N.C., SOLA - A Numerical Solution Algorithm for Transient Fluid Flows, Los Alamos Scientific Laboratory report LA-5852 (April 1975)

29. Welch, J.E., Harlow, F.H., Shannon, J.P., and Daly, B.J., The MAC Method: A Computing Technique for Solving Viscous, Incompressible, Transient Fluid - Flow Problems Involving Free Surfaces, Los Alamos Scientific Laboratory report LA-3425 (January 1969)

30. Nichols, B.D. and Hirt, C.W., J. Comput. Phys., 8(1971) 434-448.

31. Viecelli, J.A., J. Comput. Phys. 8(1971) 119-143.

32. Nichols, B.D., Hirt. C.W., and Hotchkiss, R.S., SOLA-VOF: A Solution Algorithm for Transient Fluid Flow with Multiple Free Boundaries, Los Alamos Scientific Laboratory Report LA-8355 (August 1980)

33. Chilton, C.H., Chemical Engineers' Handbook, 5th ed., Edited by R.H. Perry and C.H. Chilton, New York: McGraw - Hill Book Co. 1973.

34. Harlow, F.A. and Amsden, A.A., Fluid Dynamics, Los Alamos Scientific Laboratory Report LA-4700 (June 1971)

35. Thom, A. and Apelt, C.J., Field Computations in Engineering and Physics. Princeton, N.J. : D. Van Nostrand Co., Inc., 1961.

36. Huh, C. and Scriven, L.E., J. Colloid and Interface Sci., 35 (1971) 85-101.

37. Dussan, V., E.B. and Davis, S.H., J. Fluid Mech., 65 (1974) 71-95.

38. Ngan, C.G. and Dussan, V., E.B., J. Fluid Mech., 118 (1982) 27-40.

BIBLIOGRAPHY Cont'd

39. Huh, C. and Mason, S.G., J. Fluid Mech., 81 (1977) 401-419.

40. Rushak, K.J., J. Fluid Mech., 119 (1982) 107-120.

41. Spiers, R.P. and Wilkinson, W.L., Chem. Eng. Sci., 29 (1974) 1821-1825.

42. Coles, D., J. Fluid Mech., 21 (1965) 385-425.

43. Holodniok, M., Kubicek, M., and Hlavacek, V., J. Fluid Mech., 108 (1981) 227-240.

44. Nguyen, N.D., Ribault, J.P., and Florent, P., J. Fluid Mech., 68 (1975) 369-388.

45. Taylor, G.I., J. Fluid Mech., 16 (1963) 595-619.

46. Leonard, B.P., Computer Methods in Appl. Mech. & Eng., 19 (1979) 59-98.

47. Harlow, F.H., Numerical Methods for Fluid Dynamics, an Annotated Bibliography, Los Alamos Scientific Laboratory report LA-4281 (September 1969)

48. Langley Research Center, Numerical Studies of Incompressible Viscous Flow in a Driven Cavity. Washington, D.C.: National Aeronautics and Space Adiministration, 1975.

49. Cebeci, T., Hirsh, R.S., Keller, H.B., and Williams, P.G., Computer Methods in Appl. Mech. & Eng., 27 (1981) 13-44.

50. Yeung, R.W., Ann. Rev. Fluid Mech., 14 (1982) 395-442.

51. Emmons, H.W., Ann. Rev. Fluid Mech., 2 (1970) 15-36.

52. Orszag, S.A. and Israeli, M., Ann. Rev. Fluid Mech., (1974) 281-318.

53. Dussan, V., E.B., Ann. Rev. Fluid Mech., 11 (1979) 371-400.

54. Gutoff, E.B. and Kendrick, C.E., AIChE, J., 28 (1982) 459-466.