To The Graduate Council:

I am submitting herewith a thesis written by Albert McGuire Hines entitled "A CFD Laboratory Expert System Environment for Free Surface Hydromechanics Flow Analysis." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Engineering Science.

A J Baker

A. J. Baker, Major Professor

We have read this thesis
and recommend its acceptance:

James A. Euler

Osama Soliman

Accepted for the Council:

C W Minkel

Vice Provost
and Dean of the Graduate School

A CFD LABORATORY EXPERT SYSTEM ENVIRONMENT FOR FREE SURFACE HYDROMECHANICS FLOW ANALYSIS

A Thesis
Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Albert McGuire Hines

May 1990

DEDICATION

It is appropriate that this work be dedicated to Jesus Christ, since He both created and is the first to observe and describe free surface flow phenomena.

> *In the beginning, God created the heavens and the earth. And the earth was formless and void, and darkness was over the face of the deep. And the Spirit of God was moving over the surface of the waters.*
>
> *Gen 1:1-2*

ACKNOWLEDGEMENTS

I have been assisted by a great number of individuals in the preparation of this report. Of first importance, I am appreciative to Leslie, my beautiful wife. She has been both patient and encouraging throughout our engagement and marriage. Of course, thanks Mom, alias Martha Jane Hines for council, financial aid; in short, just for being the greatest Mom. Thanks to Maranatha Christian Church for providing a true family during these school years.

I would also like to express my sincere gratitude to Dr. A. J. Baker, my major professor, as well as my other committee members, Dr. Osama Soliman and Dr. James A. Euler, for their timely guidance and friendship. I would also like to thank Giuseppe (alias "Joe") Iannelli for his availability and eagerness to provide solutions to sometimes monumental problems. Special thanks to Dr. J. E. Stoneking, Bill Lyday, Erwin Boyce, Eunice Hinkle, Faye Muly, Berdie Parsons, Angela McCarter, Judy Dunn and the rest of the Engineering Science and Mechanics department for moral, technical and financial support over the [many] years. Finally, thanks to the Army Corps of Engineers Waterways Experiment Station for initial funding of this project.

ABSTRACT

A pilot research code has been upgraded technically and aesthetically to a fully applied hydrodynamics package. This set of five programs, entitled FATHOM, approximately solves the two-dimensional free surface hydromechanics form of the Navier-Stokes equations. Included in the program set is an automated grid generator, the ability to define various types of flow obstructions, as well as access to a wide variety of graphical input/output devices. Flexibility, accuracy, ease of use and solution stability are documented with results from six test cases chosen to examine a wide and challenging variety of applications in free-surface hydrodynamics. This report constitutes a complete user's guide for the five program system FATHOM.

The numerical approximation is implemented through a finite element spatial discretization. This yields a time dependent, first-order differential equation system. These ordinary differential equations are integrated through an implicit scheme of variable order. The finite element analysis is carried out utilizing Taylor weak statement theory to increase stability. The integration in time may be performed employing either a rectangular, trapezoidal or implicit Runge-Kutta scheme. A tensor matrix factorization allows the resulting matrix equation system to be solved economically.

A range of benchmark problems are reported examining stability, accuracy, ease of use and flexibility. Results are presented confirming that these demands are met by FATHOM. Additionally, the implementation of the tensor matrix factorization effectively reduces the amount of computation execution time drastically over traditional direct methods.

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

The CFD Laboratory

Obtaining valid solutions for practical design problems has always been the goal of the engineer. In the field of computational fluid dynamics, or CFD, this goal is being pursued through the concept of the CFD Laboratory "expert system environment." This thesis deals with implementing the finite element computer program system FATHOM, into the CFD Laboratory. FATHOM approximately solves the two-dimensional depth-averaged shallow-flow free-surface hydrodynamics equations.

The implementation leading to the laboratory environment involves several major tasks beyond creating the basic research code. The code must be executed for many different cases to verify correctness and gain familiarity with its various operational features, limitations, input/output streams and overall logic. Code refinements must then be made to better achieve the goals of the lab environment. Next, a user's guide must be prepared to document access to the program and give instructions on its use, care, nomenclature and other distinguishing features. Finally, a thorough set of benchmark test cases must be run to verify accuracy and utilization of the implemented features. The execution time, user effort, solution quality and program flexibility are investigated.

This thesis is the culmination of these efforts. It is the author's intention for this thesis to serve not only as a manual to the utilization of FATHOM, but also as a guide for incorporating other engineering analysis programs into the CFD laboratory environment.

A Perspective

The digital computer continues to offer possibilities of ever-increasing analysis capability to researchers in all branches of science and engineering. However, the productive utilization of this power and speed by the scientific community has been to some degree hindered by a general lack of willingness to conform to new methods. This is in part understandable. One of the basic tenets of engineering practice is to make do with the materials at hands, and hence, "If it works, don't fix it!" This persuasion has unfortunately often led to an aversion to assimilating the newest technology resulting in obsolescence. While the engineering profession must, by economic necessity, remain generally conservative, new and more competitive methods must be sought out and implemented to meet the challenges of the future.

Of particular interest is the use of the digital computer in fluid mechanics analysis. The equations governing fluid flow are among the most challenging in any branch of engineering. Thirty years ago, the overwhelming difficulty of analytically solving these equations for any real problem were so great that there were not more than a handful of analytical solutions available to the researcher. And almost all of these solutions were obtained by simply throwing away enough troublesome terms to yield a tractable problem.

In contrast, the CFD laboratory will use and promote the evolution of numerical approximation theory, workstation implementation and advances in microcircuitry. These open the doors to a new horizon of practical solution acquisition to this technical frontier. In the CFD laboratory, engineers could solve specific problems at hand quickly, economically and systematically, in their full complexity, by performing "numerical experiments." The obstacles and expenses of analytical accuracy, although important, are superseded by less expensive discrete approximate solutions, whose accuracy can be carefully monitored and controlled.

As the number and complexity of CFD schemes increases, so also increases the necessity of specializing tasks in order to economically create solutions in the work place. Specifically, an engineer with the ability to create design solutions is not usually the wisest choice as the terminal operator. The laboratory environment must thus be established to offer the various engineers and researchers the option of writing, developing, analyzing software or obtaining results from the produced codes.

This CFD laboratory would ideally consist of an accessible mainframe computer system networked to "intelligent" and "user-friendly" graphics workstations. At these workstations, the engineer would set up problem geometry and flow state definitions directly on the terminal. The workstation would request only the necessary engineering problem parameter data. The mainframe would then run the numerical approximation and send the results back to the workstation which would then display the results in an easily interpreted "snap-shot" presentation. This format could be easily modified by the user to produce a hard copy output as it appears on the screen.

Meanwhile, computer program development and redesign could be carried out by others working in the same lab. The strengths and weaknesses of current software could be evaluated by the operators and immediately accounted for in continual code improvement. Those developing codes would be able to instruct the users in the most effective ways to utilize the newly introduced code features. The result would be a solutions-oriented department not only for the purpose of developing programs and obtaining results, but continually upgrading existing theory and applications. This scenario constitutes the *"CFD Laboratory"* and attaining these primary goals creates the associated expert system environment.

Secondary goals of the CFD Laboratory environment are maintainability and flexibility of programs, user friendliness, both of equipment and programs, and availability of exhaustive references. Maintainability is the ability to make changes and analyze additions or refinements to codes in a straightforward, economical fashion. This necessitates modular (structured) programming, ample in-program comments and good program documentation. Flexibility is the ability to comply with or extend to a wide variety of problems in a simple manner. Many choices for grid geometry, boundary conditions, initial conditions, integration parameters, flow parameters and the like should be available to the researcher. Further, making one or a few of these changes to an existing problem should not require respecification of parameters which need no change.

User friendliness is the ability to offer the code user comfortable, brief and intuitive inputs and expository outputs. With the goal of economy and meaningful solutions in mind, the user should be queried in a manner which minimizes the time and effort expended. Both hardware and software should be tailored to this purpose. Problem parameters should be requested in a form familiar to the engineer. Preferably, the inputs should be "intelligent," that is, vary according to the particular variety of problem under investigation.

This concept of an interface program which interprets the given problem autonomously and configures the input session accordingly, is called the "expert systems" approach. Although a formal application of expert systems is beyond the scope of this thesis, one of the future goals of the CFD Laboratory environment is indeed to utilize expert systems concept as a means of reducing the work load and expertise level required of the code user.

Another aspect of user friendliness is that of offering the user intuitive input and output formats. Liberal usage of menus, hardware ornaments such as a mouse, digitizing equipment, light pens and the like, all ease the tedious tasks involved with protracted sessions at the terminal. An abundance of such gadgetry is available for a wide variety of computer systems. However, standardized implementation for actual programs has never kept pace with the large number of features offered. The lab environment should have the capability and encouragement necessary to take full advantage of these time and effort- saving devices.

The CFD Lab, then, is the combination of quality hardware, software, reference material and personnel at all levels of proficiency working together to promote learning while economically developing and perfecting tools for the solution of practical fluid dynamics problems. Hence the CFD Lab is the natural end-product of economizing complicated engineering computations by increasing user comfort, maintenance economy, and task specialization. Incorporating continuing advances in fluid dynamics through this lab environment is where the key to the future lies for effective computational fluid dynamics utilization in design engineering analysis productivity. This thesis represents a significant step in this direction for the given problem class.

CHAPTER I

THE GOVERNING EQUATIONS

Free Surface Shallow Flow Form

The equations governing unsteady, incompressible, constant viscosity, laminar, three-dimensional flows in non-dimensional and non-conservation Cartesian scalar form are the continuity of mass

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

and Navier Stokes

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} - \frac{1}{Re}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) + \frac{\partial p}{\partial x} - F_x = 0 \tag{2}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} - \frac{1}{Re}\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right) + \frac{\partial p}{\partial y} - F_y = 0 \tag{3}$$

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} - \frac{1}{Re}\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right) + \frac{\partial p}{\partial z} - F_z + \frac{1}{Fr^2} = 0 \tag{4}$$

Since gravity is assumed vertical, its horizontal contributions are not present. The following are standard variable definitions

x,y,z - Cartesian spatial coordinates, normalized with reference length L,

u,v,w - x,y,z velocity components, normalized with reference velocity U,

t - time normalized with (L/U),

p - pressure normalized with (ρU^2), ρ is fluid density

F_i - external specific force in the i^{th} direction, normalized with $(\rho U^2/L)$,

Re - Reynolds number (UL/ν), where ν is the kinematic viscosity,

Fr - local Froude number, for velocity vector magnitude $|\mathbf{u}|$,

$$Fr(x,y,t) \equiv \left(\frac{|\mathbf{u}|}{\sqrt{gh}}\right)$$

In analogy with the Mach number, $Fr > 1.0$ defines super-critical flow.

Figure 1 illustrates the geometry and notation for a constrained, shallow, free surface flow with a uniform gravitational field in the z direction. The shallow flow form of (1) - (4) is obtained by averaging u and v vertically. Define the depth average of any variable $q(x,y,z,t)$ as

$$\bar{q}(x,y,t) \equiv \frac{1}{h}\int_{z=b}^{b+h} q(x,y,z,t)\, dz \tag{5}$$

where $h(x,y,t)$ is the local depth distribution of the fluid free surface and $b(x,y)$ is the bed bathymetry.

The pertinent partial derivatives are determined using Leibnitz theorem,

$$\frac{\partial}{\partial x}(h\bar{u}) = \frac{\partial}{\partial x}\int_b^{b+h} u\, dz = \left(u\frac{\partial z}{\partial x}\right)\Bigg|_b^{b+h} + \int_b^{b+h}\frac{\partial u}{\partial x}\, dz \tag{6}$$

$$\frac{\partial}{\partial y}(h\bar{v}) = \frac{\partial}{\partial y}\int_b^{b+h} v\, dz = \left(v\frac{\partial z}{\partial y}\right)\Bigg|_b^{b+h} + \int_b^{b+h}\frac{\partial v}{\partial y}\, dz \tag{7}$$

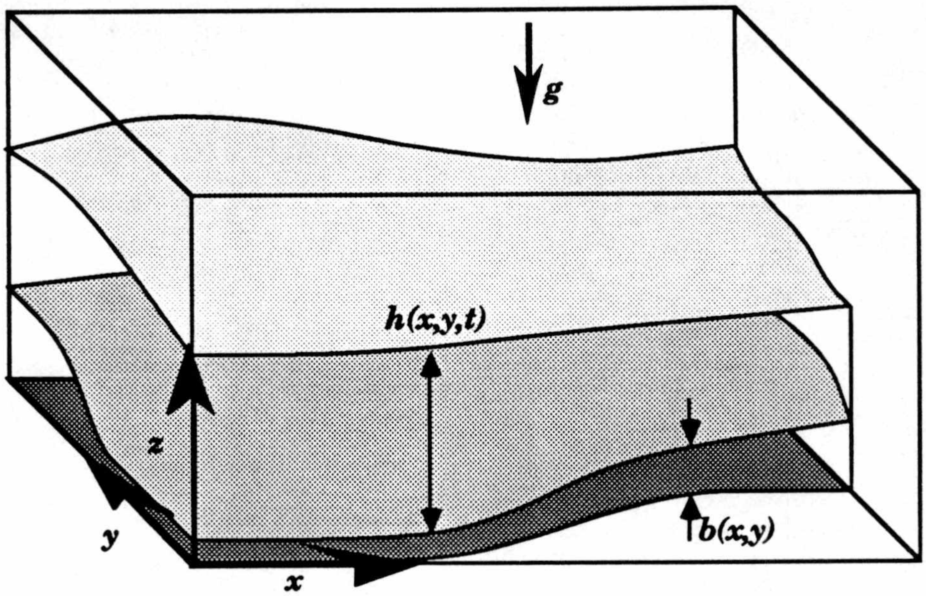

Figure 1. Geometry of a generic free surface flow situation.

where the overbar denotes the vertically integrated variable, i.e., (5). The integration is carried out from $z=b$, the channel bed, to $z=b+h$, the free surface. Summing equations (6) and (7) and using (1) yields

$$\frac{\partial}{\partial x}(h\bar{u}) + \frac{\partial}{\partial y}(h\bar{v}) =$$

$$\left[u\big|_{b+h}\frac{\partial(b+h)}{\partial x} - u\big|_b\frac{\partial b}{\partial x}\right] + \left[v\big|_{b+h}\frac{\partial(b+h)}{\partial y} - v\big|_b\frac{\partial b}{\partial y}\right] + \int_b^{b+h}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)dz$$

$$= u\big|_{b+h}\frac{\partial h}{\partial x} + v\big|_{b+h}\frac{\partial h}{\partial x} + \int_b^{b+h} -\left(\frac{\partial w}{\partial z}\right)dz + u\big|_{b+h}\left(\frac{\partial b}{\partial x}\right) + v\big|_{b+h}\left(\frac{\partial b}{\partial y}\right)$$

$$= u\big|_{b+h}\frac{\partial h}{\partial x} + v\big|_{b+h}\frac{\partial h}{\partial x} - w\big|_b^{b+h} + u\big|_{b+h}\left(\frac{\partial b}{\partial x}\right) + v\big|_{b+h}\left(\frac{\partial b}{\partial y}\right) \tag{8}$$

The term $w\big|_b^{b+h}$, in equation (8) is the difference between the free surface and bed z-component of velocity. Thus

$$w\Big|_b^{b+h} = \frac{d(b+h)}{dt}\Big|_{b+h} - \frac{db}{dt}\Big|_b = \left(\frac{dh}{dt} + \frac{db}{dt}\right)\Big|_{b+h} - \frac{db}{dt}\Big|_b \tag{9}$$

Expanding (9)

$$w\Big|_b^{b+h} = \frac{\partial h}{\partial t} + u\Big|_{b+h}\frac{\partial h}{\partial x} + v\Big|_{b+h}\frac{\partial h}{\partial y} + \frac{\partial b}{\partial t} + u\Big|_{b+h}\frac{\partial b}{\partial x} + v\Big|_{b+h}\frac{\partial b}{\partial y}$$

$$- \frac{\partial b}{\partial t} - u\Big|_b\frac{\partial b}{\partial x} - v\Big|_b\frac{\partial b}{\partial y}$$

$$= \frac{\partial h}{\partial t} + u\Big|_{b+h}\frac{\partial h}{\partial x} + v\Big|_{b+h}\frac{\partial h}{\partial y} + u\Big|_{b+h}\frac{\partial b}{\partial x} + v\Big|_{b+h}\frac{\partial b}{\partial y} - u\Big|_b\frac{\partial b}{\partial x} - v\Big|_b\frac{\partial b}{\partial y} \tag{10}$$

by the definition of the substantial derivative. The partial derivative of b with respect to time vanishes for a stationary bed. Inserting (10) into (8) yields

$$\frac{\partial}{\partial x}(h\bar{u}) + \frac{\partial}{\partial y}(h\bar{v}) = -\frac{\partial h}{\partial t} \tag{11}$$

Thus the replacement of the continuity equation (1) for the depth-averaged shallow flow formulation is

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(hu) + \frac{\partial}{\partial y}(hv) = 0 \tag{12}$$

where the overbar notation is now deleted without ambiguity. In fully expanded form (12) becomes

$$\frac{\partial h}{\partial t} + h\frac{\partial u}{\partial x} + h\frac{\partial v}{\partial y} + u\frac{\partial h}{\partial x} + v\frac{\partial h}{\partial y} = 0 \tag{13}$$

The applicable depth-averaged x and y momentum forms are similarly derived from (2) - (3). If the velocity , w in (4) is negligible in comparison to u

and v (i.e., modest bed slopes) and no external tractions are present in the vertical direction, then (4) reduces to the hydrostatic statement

$$\frac{\partial p}{\partial z} = -\frac{1}{Fr^2} \quad , \quad p = -\frac{z}{Fr^2} + C_o \tag{14}$$

Evaluating the gauge pressure $P_g|_{z=b+h} = P - P_{atm} = 0$ at the free surface

$C_o = \dfrac{h+b}{Fr^2}$ and hence

$$p = \left(\frac{h+b-z}{Fr^2}\right) \tag{15}$$

Assuming the only external traction in the force vector F_i is associated with the bed drag, the following expression is obtained [9]

$$F_i = \frac{u_i}{hC_g^2}|\mathbf{u}| \tag{16}$$

where C_g is the the non-dimensional Chezy coefficient, [4]

$$C_g = \frac{C}{\sqrt{g}} = \sqrt{\frac{8}{f}} = \sqrt{\frac{8}{\left(\frac{8\tau_b}{\rho U^2}\right)}} = \sqrt{\frac{\rho U^2}{\tau_b}} \tag{17}$$

C is the Chezy constant, f is the Darcy friction factor and τ_b the shear stress at the bed [9]. Multiplying (2) by h, (13) by u and adding yields

$$h\frac{\partial u}{\partial t} + \left(u\frac{\partial h}{\partial t} + hu\frac{\partial u}{\partial x} + hu\frac{\partial v}{\partial y} + u^2\frac{\partial h}{\partial x} + vu\frac{\partial h}{\partial y}\right) + hu\frac{\partial u}{\partial x} + hv\frac{\partial u}{\partial y}$$

$$- \frac{h}{Re}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + h\frac{\partial(h+b)}{Fr^2 \partial x} - hF_x = 0 \tag{19}$$

utilizing the hydrostatic pressure relation (15). Combining into exact differentials and separating the pressure term,

$$\frac{\partial(hu)}{\partial t} + \frac{\partial(hu^2)}{\partial x} + \frac{\partial(hvu)}{\partial y} - \frac{h}{Re}\nabla^2 u + \frac{h}{Fr^2}\frac{\partial h}{\partial x} - hF_x + \frac{h}{Fr^2}\frac{\partial b}{\partial x} = 0 \tag{20}$$

the x momentum equation then becomes

$$\frac{\partial}{\partial t}(hu) + \frac{\partial}{\partial x}(hu^2) + \frac{\partial}{\partial y}(huv) + \frac{1}{Fr^2}\left(\frac{\partial}{\partial x}\left(\frac{h^2}{2}\right)\right) + \frac{1}{Fr^2}\left(h\frac{\partial b}{\partial x}\right)$$

$$- \frac{h}{Re}\nabla^2 u - \frac{u}{C_g^2}|\mathbf{u}| = 0 \tag{21}$$

Similarly, for the y momentum equation

$$\frac{\partial}{\partial t}(hv) + \frac{\partial}{\partial x}(huv) + \frac{\partial}{\partial y}(hv^2) + \frac{1}{Fr^2}\left(\frac{\partial}{\partial y}\left(\frac{h^2}{2}\right)\right) + \frac{1}{Fr^2}\left(h\frac{\partial b}{\partial y}\right)$$

$$- \frac{h}{Re}\nabla^2 v - \frac{v}{C_g^2}|\mathbf{u}| = 0 \tag{22}$$

Define the variables m_x and m_y as the volume flux per unit area, i.e.,

$$m_x \equiv uh \tag{23}$$

$$m_y \equiv vh \tag{24}$$

and insert these into (19)-(22). Using tensor indicial notation yields the final form of the two-dimensional depth-averaged shallow flow equations as

$$\frac{\partial}{\partial t}(h) + \frac{\partial}{\partial x_i}(m_j) = 0 \tag{25}$$

$$\frac{\partial m_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\frac{m_j m_i}{h}\right) + \frac{1}{Fr^2}\left(\frac{\partial}{\partial x_i}\left(\frac{h^2}{2}\right)\right) + \frac{1}{Fr^2}\left(h\frac{\partial b}{\partial x_i}\right) - \frac{h}{Re}\nabla^2\left(\frac{m_i}{h}\right)$$
$$- \frac{1}{C_g}\left(\frac{m_i(m_j m_j)^{\frac{1}{2}}}{h^2}\right) = 0 \tag{26}$$

Boundary Conditions

A characteristics analysis developed in [5] is applied to the inviscid form ($Re \Rightarrow \infty$) of (25) - (26). The results show that for a mathematically well-posed problem, the number of boundary conditions is

For sub-critical flow ($Fr < 1.0$)

inlet - specify two conditions

outlet - specify one condition

solid boundary - specify one condition

For super-critical flow ($Fr > 1.0$)

inlet - specify three conditions

outlet - specify no conditions

solid boundary - specify one condition

Specification of both h and m_x is not permissible for sub-critical flows as determined using the method of Chakravarthy [5]. It is mathematically admissible and operationally preferable that the outlet boundary condition for sub-critical flows be average hydrostatic pressure. This choice permits the CFD algorithm to autonomously adjust the depth solution across the outlet, hence lessening the need for an extended "far-field" region.

Quasi One Dimensional Analysis

An analytically useful subset of equations (25) - (26) is obtained by averaging in the transverse horizontal (y) direction, see Figure 1. The result is a flow influenced by the cross sectional area variation of the channel, bed slope, the hydrostatic pressure and the inlet and outlet boundary conditions. This quasi one dimensional form is completely analogous to the renowned isentropic and normal shock flow equations in compressible gas dynamics. The analysis gives an excellent picture of the overall behavior for many practical channel flows while requiring a relatively small amount of computational effort and cost. The quasi one dimensional solution also provides a convenient and appropriate initial condition generation procedure for the general two-dimensional problem statement.

Recalling (7), averaging in the y direction and neglecting viscous terms

$$\frac{\partial}{\partial x}(A\,\bar{m}_x) = \left(m_x \frac{\partial y}{\partial x}\right)\Big|_1^2 + \int_1^2 \frac{\partial m_x}{\partial x} dy = \left(m_x \frac{\partial y}{\partial x}\right)\Big|_1^2 + \int_1^2 -\left(\frac{\partial h}{\partial t} + \frac{\partial m_y}{\partial y}\right) dy$$

$$= m_x\Big|_1^2 \frac{\partial A}{\partial x} + \frac{\partial}{\partial t}\int_1^2 -\,h dy + \int_1^2 -\frac{\partial m_y}{\partial y} dy = m_x\Big|_1^2 \frac{\partial A}{\partial x} - \frac{\partial (A\bar{h})}{\partial t} - m_y\Big|_1^2 \qquad (27)$$

where A, the channel width, represents y evaluated from 1 to 2, i.e., the difference in y-coordinates of the two sides of the river bank. The term Ah occurring in the next to last term in (27) is derived from the definition of the width-averaged free surface height,

$$\frac{\partial \bar{h}}{\partial t} = \frac{\partial}{\partial t}\left(\frac{1}{A}\int_1^2 h\,dy\right) \tag{28}$$

since the bank profile is not a function of time. The last term in (27) remains to be evaluated. The y component of momentum at the wall boundaries may be derived directly from geometrical considerations or in terms of the definition of the variables

$$m_y\Big|_1^2 = hv\Big|_1^2 = h\frac{dy}{dt}\Bigg|_1^2 = h\left(\frac{dy}{dx}\frac{dx}{dt}\right)\Bigg|_1^2 = h\frac{dy}{dx}u\Bigg|_1^2 = \frac{dA}{dx}m_x\Bigg|_1^2 \tag{29}$$

Utilizing (29) in (27) yields

$$\frac{\partial}{\partial x}(A\bar{m}_x) = \left(m_x\frac{\partial A}{\partial x}\right)\Bigg|_1^2 + \frac{\partial A\bar{h}}{\partial t} - m_x\frac{dA}{dx}\Bigg|_1^2 = \frac{\partial A\bar{h}}{\partial t} = A\frac{\partial \bar{h}}{\partial t} \tag{30}$$

which may also be expressed as

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(m) + m\left(\frac{d\,ln(A)}{dx}\right) = 0 \tag{31}$$

The momentum equation is derived in a manner similar to (19) yielding

$$\frac{\partial m}{\partial t} + \frac{\partial}{\partial x}\left(\frac{m^2}{h}\right) + \frac{1}{Fr^2}\frac{\partial}{\partial x}\left(\frac{h^2}{2}\right) + \frac{h}{Fr^2}\frac{\partial b}{\partial x} + \frac{m^2}{h}\left(\frac{d\,ln(A)}{dx}\right) + \frac{m|m|}{C_g^{\,2}h^2} = 0 \tag{32}$$

Equations (31) - (32) may be solved analytically for steady inviscid flow. Directly integrating the continuity equation (31) yields

$$m\ ln(A) = \text{constant} = uh\ ln(A) \tag{33}$$

Expanding (31)-(32), employing (33) and the exact differential yields

$$h\frac{\partial u}{\partial x} + u\frac{\partial h}{\partial x} = -\ uh\frac{\partial \ln(A)}{\partial x} \tag{34}$$

$$2uh\frac{\partial u}{\partial x} + u^2\frac{\partial h}{\partial x} + \frac{h}{Fr^2}\frac{\partial h}{\partial x} + \frac{h}{Fr^2}\frac{\partial b}{\partial x} = -\ u^2h\frac{\partial \ln(A)}{\partial x} \tag{35}$$

Substituting (35) into (34) yields

$$u\frac{\partial u}{\partial x} + g\frac{\partial h}{\partial x} + g\frac{\partial b}{\partial x} = 0 \tag{36}$$

Bringing all terms inside the derivative yields

$$\frac{\partial}{\partial x}\left(\frac{u^2}{2} + gh + gb\right) = 0 \tag{37}$$

Hence,

$$\frac{u^2}{2} + gh + gb = g(h_o + b_o) = g(h + b)_o \tag{38}$$

where subscript zero denotes the stagnation reference state for fixed datum. Rewriting in terms of the local Froude number *Fr*, local channel width *A(x),* and the local velocity respectively, this becomes [6]

$$\frac{h}{H_o} = \left(1 + \frac{Fr^2}{2}\right)^{-1} \qquad (39)$$

$$\frac{A}{A^*} = \frac{1}{Fr}\left[\frac{2}{3}\left(1 + \frac{Fr^2}{2}\right)\right]^{\frac{3}{2}} \qquad (40)$$

$$\frac{u}{u^*} = \frac{\sqrt{6}}{2} Fr\left(1 + \frac{Fr^2}{2}\right)^{-\frac{1}{2}} \qquad (41)$$

where the asterisk represents the critical state (at $Fr = 1.0$) in analogy with compressible gas dynamics. Given the inlet Fr and $A(x)$, the velocity and free surface height may thus be determined immediately for the entire channel in the steady quasi-one dimensional simplification.

Quasi 1-D Flow Boundary Conditions

When specifying boundary conditions for quasi one-dimensional flows, it is necessary to specify the width distribution $A(x)$, the inlet flow Froude number Fr, and either the inlet height or velocity. From equation (40), the Froude number distribution may then be determined. And, by equation (39) or (41), the free surface height or velocity distribution may be obtained.

CHAPTER II

THE CFD NUMERICAL PROCEDURE

Generalized Coordinate System Description

Equations (25) - (26) may be expressed as

$$L(q) = \frac{\partial q}{\partial t} + \frac{\partial}{\partial x_k}\left(f_k - f_k^v\right) - s = 0 \tag{42}$$

where q is the state variable array of dependent variables, s is the source term array, f_k^v is the viscous flux vector, i.e. Reynolds number term in equation (26), f_k is the remainder of the flux terms, and k ranges from 1 to 2. The flux vector f_k includes the kinetic and kinematic contributions. In the two dimensional formulation, the source term includes the bed depth and bed drag terms. The quasi one-dimensional analysis adds the channel width distribution as a source term.

The conservation law system (42) is written in terms of the cartesian variables x and y. It is desirable to reexpress (42) in general curvilinear coordinates in order to readily handle irregular boundary geometries. This transformation is accomplished by mapping the Cartesian x and y coordinates to the general curvilinear coordinates η_1 and η_2. The Jacobian of the transformation

$$x_i = x_i(\eta_1, \eta_2) \tag{43}$$

is

$$J = \begin{bmatrix} \left(\frac{\partial x}{\partial \eta_1}\right), \left(\frac{\partial y}{\partial \eta_1}\right) \\ \left(\frac{\partial x}{\partial \eta_2}\right), \left(\frac{\partial y}{\partial \eta_2}\right) \end{bmatrix} \tag{44}$$

The spatial derivatives in (42) are then evaluable as

$$\left(\frac{\partial[\bullet]}{\partial x_i}\right) = \frac{1}{det(J)} e_{ij} \left(\frac{\partial[\bullet]}{\partial \eta_j}\right) \tag{45}$$

where $det(J)$ is the determinant of the Jacobian J in (44) and e_{ij} is the i^{th} row and j^{th} column entry of the co-factor matrix of $det(J) \bullet J^{-1}$. Thus, (42) becomes

$$L(q) = \frac{\partial q}{\partial t} + \frac{e_{jk}}{det(J)} \frac{\partial f_j}{\partial \eta_k} - \frac{e_{jk}}{det(J)} \frac{\partial f_j^v}{\partial \eta_k} - s = 0 \tag{46}$$

For any coordinate transformation, the fundamental invariance relation is

$$\frac{\partial e_{ij}}{\partial \eta_j} = 0 \tag{47}$$

and for stationary meshes $det(J)$, henceforth abbreviated $|J|$, may be brought inside the time derivative. Hence (46) becomes [7]

$$L(q) = \frac{\partial q|J|}{\partial t} + \frac{\partial (e_{jk} f_j)}{\partial \eta_k} - \frac{\partial (e_{jk} f_j^v)}{\partial \eta_k} - s|J| = 0 \tag{48}$$

The Weak Statement CFD Algorithm

Equation (48) may be recast into a computationally convenient form by employing a so-called "weak statement". For any arbitrary weight function $w(\boldsymbol{x})$, a general weak statement for (48) is [1]

$$\int_{\Omega} w L(q)\; d\eta \equiv 0 \tag{49}$$

over the transformed domain Ω spanned by (η_1, η_2). The preferred weak statement form is obtained upon applying the Green-Gauss theorem to (49). Two advantages are gained thereby. First, the order of highest derivatives required of the trial space is reduced by one. Additionally, boundary conditions can be implemented in a straightforward, grid-independent manner via the generated surface integrals.

Inserting (48) for $L(q)$ with the Green Gauss application, (49) becomes

$$\int_{\Omega} w\left(\frac{\partial q|J|}{\partial t}\right) d\eta - \int_{\Omega} e_{jk}\frac{\partial w}{\partial \eta_k}\left(f_j - f_j^{v}\right) d\eta - \int_{\Omega} w s|J|\, d\eta$$

$$+ \oint_{\partial\Omega} w e_{jk}\left(f_j - f_j^{v}\right) n_k\, d\Gamma = 0 \tag{50}$$

where Ω spans the flow domain and n_j is the outward pointing unit vector normal to the domain boundary Γ.

The Taylor Weak Statement

The partial differential equation system (25) - (26) exhibits an initial value character. The semi-discrete time evolution of (50) may be determined using an implicit form of a Taylor series (Appendix 2), yielding

$$q_{n+1} = q_n + \Delta t\left((1-\theta)\frac{\partial q_n}{\partial t} + \theta\frac{\partial q_{n+1}}{\partial t}\right) \tag{51}$$

Here, n is the time step index and $0 \leq \theta \leq 1$ is the implicitness parameter, prescribing the point around which the Taylor series is performed. For example, $\theta = 0$ takes the evaluation point to be the previous time station, hence (51) is fully explicit, whereas $\theta=1$ takes the point to be the new time station, i.e. fully implicit.

By constructing the Taylor series (51) it may be determined that the differential equation system (25) - (26) numerically solved differs from the continuum governing equations by temporal truncation errors. By subtracting this difference, the semi-discrete equation set admits a solution more nearly mirroring the continuum solution. The inclusion of the highest order truncation terms is equivalent to inserting dissipative operators which can offset the dispersive error terms intrinsic to a spatially semi-discrete approximation to the continuum governing equation. Employing an adequate spatial discretization, these terms are very frequency selective and preferentially dampen the high frequency dispersion error waves while retaining the lower frequency waves associated with the various characteristics of the analytical solution. This effective mechanism for dispersive error control is discussed thoroughly by Baker and Kim [3].

Semi-discrete numerically generated dispersion errors are manifest as approximate solution oscillations of principle wavelength $2h$, where h is a length measure of the local finite element mesh. The added Taylor Weak Statement (*TWS*) terms are especially adept at filtering these extraneous oscillations when the mesh is inadequate for fully resolving the local solution gradients. Because these undesirable oscillations can corrupt the entire solution process, the *TWS* has proven to be a powerful tool for enhancing stability on the way to generating adequate mesh solutions, hence overall solution quality. The *TWS* altered governing equation, denoted $L^m(q)$, becomes [7,8]

$$L^m(q) \equiv L(q) - \beta_1\left(\frac{\partial}{\partial \eta_1}|\mathbf{u}|\frac{\partial q}{\partial \eta_1}\right) - \beta_2\left(\frac{\partial}{\partial \eta_2}|\mathbf{u}|\frac{\partial q}{\partial \eta_2}\right) \tag{52}$$

where β_i is the directional coefficient setting the level of numerical dissipation. Further, $|\mathbf{u}|$ is the velocity magnitude defined as [7]

$$|\mathbf{u}| \equiv \left(e_{jk}\left(\frac{m_k}{h}\right)e_{jl}\left(\frac{m_l}{h}\right)\right)^{\frac{1}{2}} \tag{53}$$

Hence the Taylor weak statement for (25) - (26) is

$$\int_\Omega w\left(\frac{\partial q|J|}{\partial t}\right)d\eta - \int_\Omega e_{jk}\frac{\partial w}{\partial \eta_k}\left(f_j - f_j^v\right)d\eta - \int_\Omega w|J|s\,d\eta$$

$$+ \oint_{\partial\Omega} w\, e_{jk}\left(f_j - f_j^v\right)n_k\,d\Gamma - \int_\Omega w\,\beta_r\frac{\partial}{\partial \eta_r}\left(|\mathbf{u}|\frac{\partial q}{\partial \eta_r}\right)d\eta = 0 \tag{54}$$

and the tensor indices j, k and r range 1 to 2. Note that equation (54) is an exact continuum expression for (25) - (26) for $\Delta t \rightarrow 0$ (i.e., β=0), which may then be modified (by setting $\beta \neq 0$) to account for the anticipated spatial semi-discretization approximation dispersive errors.

The Spatial Semi-Discretization

Having re-expressed the governing equation set as the Taylor weak statement, an effective *CFD* procedure is sought to generate an approximate solution to (54). Assuming space and time separable, any approximation q^N to q can be expressed as an N-term finite series

$$q(\eta_j, t) \approx q^N(\eta_j, t) = \sum_{i=1}^{N} a_i(t) \Phi_i(\eta_j) \tag{55}$$

where $a_i(t)$ is the set of time-dependent unknown expansion coefficients for the selected set of functions $\Phi_i(\eta_j)$, called the "trial space." The approximation q^N will be inserted into the governing TWS, hence the coefficients a_i will eventually be determined to complete the approximation.

Following Galerkin's method of weighted residuals, the weight function set $w_i(\eta_j)$ is chosen to be identical to the trial space set $\Phi_i(\eta_j)$. This assures an optimal solution in the sense that the approximation error is required to be orthogonal to the specific trial function set chosen. Hence, the generic weak statement (49) becomes [1], for (54),

$$TWS = \int_{\Omega} \Phi_i(\eta_j) L^m \left(\sum_{i=1}^{N} a_i(t) \Phi_i(\eta_j) \right) d\eta = 0 \tag{56}$$

where N is the number of terms chosen for (55). Upon defining the specific trial function set Φ_i, the only remaining unknowns are the coefficients $a_i(t)$.

The finite element method suggests that the $\Phi_i(\eta_j)$ be constructed as piecewise continuous sets of polynomials of at least degree one. Piecewise continuous polynomials of low order permit a simple trial space construction and the potential to approximate the solution as accurately as desired by increasing the number of polynomial "pieces." The final, or global solution is the assembly of these individual segments, each being defined on an element. The following investigates one such typical "elemental" solution.

Experience and theory has shown the expedient choice for the trial space polynomial set is the union of Lagrange interpolation polynomials. One convenient feature of these "basis" functions is that the values of the unknown coefficients correspond to the numerical value of the approximate solution at some particular locations, called nodes, within the element. Representing the column matrix of basis functions as $\{N\}$, and the unknowns $a_i(t)$ on an element as $\{Q(t)\}_e$ the finite element expression for the approximation variable q^N on a finite element domain Ω_e is

$$q(\eta_j, t) \approx q_e(\eta_j, t) = \{N_k(\eta_j)\}^T \{Q(t)\}_e \tag{57}$$

where the subscript "e" denotes pertaining to the finite element Ω_e. The number of entries in the two matrices are dependent upon the degree k of the (Lagrange) polynomial chosen, and generally equals the number of geometric nodes in the finite element. The particular choice of $\{N_k\}$ in the FATHOM code is the bi-linear quadrilateral basis containing four nodes per element. Thus,

$$\{N_1^+\} = \frac{1}{4}\begin{Bmatrix} (1-\eta_1)(1-\eta_2) \\ (1+\eta_1)(1-\eta_2) \\ (1+\eta_1)(1+\eta_2) \\ (1-\eta_1)(1+\eta_2) \end{Bmatrix} \tag{58}$$

where η_1 and η_2 are the curvilinear coordinate system components spanning the generic finite element with origin at the centroid (see Figure 2).

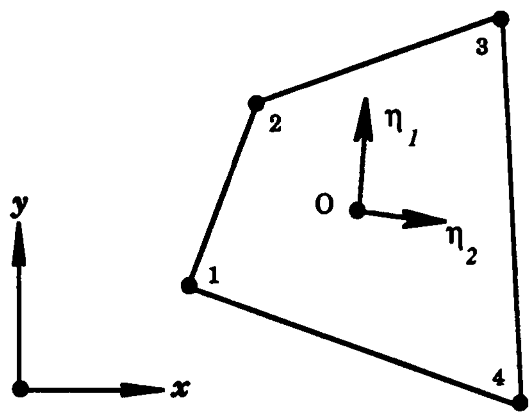

Figure 2. The local coordinate system η_i on a finite element Ω_e.

The basis function $\{N_k(\eta_i)\}$ is defined in terms of the local coordinates η_1 and η_2 which span the element. However, the *TWS* (54) contains the terms e_{jk} and $|J|$ which are expressed in terms of the global Cartesian coordinates x and y. The convenient resolution is the isoparametric transformation [1] of coordinates spanning Ω_e, where the global variables are interpolated in terms of the nodal coordinates of the basis functions used for the trial space,i.e.,

$$x_e(\eta_1, \eta_2) \equiv \left\{N_1^+(\eta_i)\right\}^T \{X\}_e \tag{59}$$

$$y_e(\eta_1, \eta_2) \equiv \left\{N_1^+(\eta_i)\right\}^T \{Y\}_e \tag{60}$$

Here, $\{X\}_e$ and $\{Y\}_e$ are column matrices of corresponding nodal coordinates.

All terms in (54) are now evaluable for any, i.e., the generic, finite element. The global approximate solution q^h is now expressed as the union of all element solutions, i.e.

$$q \approx q^h = \bigcup_e q_e = \bigcup_e \left(\left\{N(\eta_i)\right\}^{\mathrm{T}} \{Q(t)\}_e\right) \tag{61}$$

Writing out (56) in full detail using (58) yields the semi-discrete *TWS*

$$S_e\Big(\int_\Omega \{N\}\{N\}^T \, d\eta \{Q|J|\}_e' - EJK_e \int_\Omega \Big(\frac{\partial\{N\}}{\partial\eta_k}\Big)\{N\}^T \, d\eta (\{FJ\}_e - \{FVJ\}_e)$$

$$- \int_\Omega \{N\}\{N\}^T d\eta\{S|J|\}_e + \int_\Omega \Big\{(\beta|u|)_k\Big\}_e^T \{N\} \frac{\partial\{N\}}{\partial\eta_k}\frac{\partial\{N\}}{\partial\eta_k}^T d\eta\{Q\}_e$$

$$+ EJK_e \oint_{\partial\Omega} \{N\}\{N\}^T d\Gamma(\{FJ\}_e - \{FVJ\}_e)\, n_k \Big) = \{0\} \tag{62}$$

where S_e is the so-called "assembly operator" denoting matrix superposition of the contributions from each of the finite elements to the global statement [1]. The superscript prime denotes the derivative with respect to time. A single evaluation of the metric data distribution e_{jk} is performed for each element at an optimally accurate coordinate pair [7,8] and assumed constant over that element. It is taken outside of the integral and renamed EJK_e. This represents

$$EJK_e \equiv \left(det(J) \cdot J^{-1}\right)_e = \text{cofactor}\begin{bmatrix} \left(\frac{\partial x}{\partial\eta_1}\right), \left(\frac{\partial y}{\partial\eta_1}\right) \\ \left(\frac{\partial x}{\partial\eta_2}\right), \left(\frac{\partial y}{\partial\eta_2}\right) \end{bmatrix}_e$$

$$= \begin{bmatrix} \left(\frac{\partial\{N\}}{\partial\eta_2}\{Y\}_e\right), & -\left(\frac{\partial\{N\}}{\partial\eta_1}\{Y\}_e\right) \\ -\left(\frac{\partial\{N\}}{\partial\eta_2}\{X\}_e\right), & \left(\frac{\partial\{N\}}{\partial\eta_1}\{X\}_e\right) \end{bmatrix}_e \tag{63}$$

Imposition of Boundary Conditions

Imposition of the boundary conditions is straightforward via the last term in (62). For wall tangency and homogeneous Neumann boundary conditions, this surface integral is simply deleted. Dirichlet constraints (as occur at inlets and outlets) are imposed by replacing one or more of the equations in the matrix algebra system with the known prescribed values.

It has been determined [8] that deleting the surface integral to enforce tangency boundary conditions is a necessary, but insufficient condition to secure continuity of mass at geometric corner singularities of a domain. The time derivative of momentum flux across the boundary is also required to vanish. Hence, for a practical implementation the flow out of or into such corners is enforced to zero for all time.

The Time Integration Procedure

Equation (62) defines a semi-discrete (that is discrete in space yet still continuous in time) matrix equation involving ordinary time derivatives. Equation (62) may be concisely written as,

$$M\{Q\}' + \{R(Q)\} = \{0\} \tag{64}$$

where the residual $\{R\}$ is a function of $\{Q\}$, the prime denotes a first derivative in time, with the mass matrix M coupling these derivatives. A procedure is sought to solve for a transient evolution of $\{Q\}$ with (64) providing the temporal derivative. The variably-implicit Taylor series is again utilized, hence

$$F(\{Q\}) \equiv M\left(\{Q\}_{n+1} - \{Q\}_{n}\right) + \Delta t\left(\theta\{R\}_{n+1} + (1-\theta)\{R\}_{n}\right) = \{0\} \tag{65}$$

Using Newton's method

$$-\frac{\partial F}{\partial\{Q\}}\left(\{Q\}^{p+1}-\{Q\}^{p}\right)=F\{Q\}^{p} \tag{66}$$

where p is the iteration index at time t^{n+1}. Equation (66) may be written in detail by evaluating the first term of (65), i.e.,

$$\frac{\partial F}{\partial\{Q\}}=M+\Delta t\theta\left(\frac{\partial\{R\}}{\partial Q}\right) \tag{67}$$

The fully discrete form of the implicit algorithm is

$$\left[M+\theta\Delta t\left(\frac{\partial R^{1}}{\partial Q}+\frac{\partial R^{2}}{\partial Q}\right)\right]\left[Q_{p+1}^{n+1}-Q_{p}^{n+1}\right]=M\left\{Q_{p+1}^{n+1}-Q^{n}\right\}$$

$$-\Delta t\left[\theta\left(R^{1}\left\{Q_{p+1}^{n+1}\right\}+R^{2}\left\{Q_{p+1}^{n+1}\right\}\right)+(1-\theta)\left(R^{1}\{Q^{n}\}+R^{2}\{Q^{n}\}\right)\right] \tag{68}$$

where $n+1$ denotes the current time index and $p+1$ is the current iteration level. R^1 and R^2 are the η_1 and η_2 direction flux terms, respectively of the residual R.

Efficient Matrix Solution Procedure

Instead of using equation (68) as is, a tensor matrix product (TMP) factorization scheme [7] is employed. The Newton Jacobian is factored into a pair of block tri-diagonal matrices. Hence the computational numerical linear algebra statement becomes:

$$\left[M_1 + \theta\Delta t\left(\frac{\partial R^1}{\partial Q}\right)\right]\{Z\} = -\{RHS\,(68)\} \tag{69}$$

$$\left[M_2 + \theta\Delta t\left(\frac{\partial R^2}{\partial Q}\right)\right]\left\{Q^{n+1}_{p+1} - Q^{n+1}_{p}\right\} = \{Z\} \tag{70}$$

where the matrices in (69) - (70) are evaluated one dimensionally. By factoring the *LHS* of (68) in this manner, the $\{Z\}$matrix solved in (69) hence, inserted into (70), allows the updated values of $\{Q\}$ for iterate p+1 to be obtained.

With this *TMP* formulation the amount of computational execution time is drastically reduced over Gaussian elimination, full matrix inversion and multiplication, or other direct methods. Appendix H shows that less than 4% of the CPU time is spent solving the matrix algebra statement (69) - (70) for a typical problem with over 13,500 degrees of freedom.

The general numerical procedure for approximating solutions to the two dimensional depth averaged, shallow flow hydrodynamics equations is now complete. Some additional details now follow for exemplification.

Quasi-One Dimensional Finite Element Formulation

The quasi-one dimensional form of (25) - (26) i.e. (31) - (32) may be written

$$L(q) = \frac{\partial q}{\partial t} + \frac{\partial f}{\partial x} - s = 0 \tag{71}$$

for the definitions

$$
\left.
\begin{aligned}
q &= \begin{Bmatrix} h \\ m \end{Bmatrix} \\
f &= \begin{Bmatrix} m \\ \left(\dfrac{m^2}{h}\right) + \left(\dfrac{h^2}{2Fr^2}\right) \end{Bmatrix} \\
s &= \begin{Bmatrix} m\left(\dfrac{d\, ln(A)}{dx}\right) \\ \dfrac{h}{Fr^2}\left(\dfrac{db}{dx}\right) + \left(\dfrac{m^2}{h}\left(\dfrac{d\, ln(A)}{dx}\right)\right) + \dfrac{m|m|}{C_g^{\,2} h^2} \end{Bmatrix}
\end{aligned}
\right\}
\tag{72}
$$

The *TWS* algorithm statement for the continuity equation of the quasi one-dimensional simplification is

$$
\begin{aligned}
S_e\Bigg(&\int_\Omega \{N\}\{N\}^T \, d\eta \{|J|H\}_e' - l_e \int_\Omega \frac{\partial\{N\}}{\partial\eta}\{N\}^T \, d\eta\{M\}_e \\
&+ \int_\Omega \frac{\partial\{N\}}{\partial x}\{N\}^T \, d\eta \left\{|J|\, M \frac{\partial\, ln(A)}{\partial x}\right\}_e - \{\beta_r|\mathbf{u}|\}_e^T \int_\Omega \{N\} \frac{\partial\{N\}}{\partial\eta}\frac{\partial\{N\}}{\partial\eta}^T d\eta\{H\}_e \\
&+ \oint_{\partial\Omega} \{N\}\{N\}^T d\Gamma\{MDOTN\} \Bigg) = \{0\}
\end{aligned}
\tag{73}
$$

where l_e is the local finite element length and $\{MDOTN\}$ is the array of discrete values of $m \bullet n$, which equal m at an outlet, $(-m)$ at an inlet, and zero everywhere else. The *TWS* momentum equation form is

$$S_e\Big(\int_\Omega \{N\}^T\{N\}\, d\eta \{|J| M\}_e^{\prime} - l_e \{M/H\}_e^T \int_\Omega \{N\}\frac{\partial\{N\}}{\partial\eta}\{N\}^T\, d\eta \{M\}_e$$

$$-\left(\frac{l_e}{2Fr^2}\right)\{H\}_e^T \int_\Omega \{N\}\frac{\partial\{N\}}{\partial\eta}\{N\}^T\, d\eta \{H\}$$

$$+\int_\Omega \{N\}^T\{N\}\, d\eta |J| \left(\left\{ \frac{H}{Fr^2}\left(\frac{db}{dx}\right)\right\}_e + \left\{\frac{M^2}{H}\frac{\partial\, ln(A)}{\partial x}\right\}_e + \left\{\frac{|M|M}{C_g{}^2 H^2}\right\}_e \right)$$

$$-\{\beta_r D\}_e^T \int_\Omega \{N\}\left(\frac{\partial\{N\}}{\partial\eta}\right)\frac{\partial\{N\}^T}{\partial\eta} d\eta \{M\}_e + \oint_{d\Omega} \{N\}^T\{N\} d\Gamma \left\{\frac{H^2\, DOTN}{2Fr^2}\right\}_e$$

$$+\frac{1}{|J|}\left(\frac{1}{Re}\right)\oint_{d\Omega}\{N\}^T\{N\}\left(\frac{d\{N\}^T}{d\eta}\right) d\Gamma \left\{\left(\frac{M^2}{H}\right) DOTN\right\}_e \Big) = \{0\} \tag{74}$$

These may be compactly written in terms of a standardized finite element matrix nomenclature [1] as

$$S_e\Big([A200]\{|J|H\}_e^{\prime} + [A210]\{M\}_e + [A210]\left\{|J| M \frac{d\, ln(A)}{dx}\right\}_e$$

$$- [A3011]\{H\}_e + \begin{bmatrix} -\delta_{e1} & 0 \\ 0 & \delta_{em} \end{bmatrix} \{M\}_e \Big) = \{0\} \tag{75}$$

where δ_{ei} is the Kronecker delta and m is the number of finite elements in the discretization. For the momentum equation,

$$S_e\left([A200]\{|J|M\}_e' - l_e\{M/H\}_e^T[A3010]\{M\}_e\right.$$

$$-\frac{l_e}{2Fr^2}\{H\}_e^T[A3010]\{H\}_e - \{\beta_r D\}_e^T[A3011]\{M\}_e$$

$$+[A200]\left(\left\{\frac{|J|H}{Fr^2}\frac{db}{dx}\right\}_e + \left\{\frac{|J|M^2}{H}\frac{dA}{dx}\right\}_e + \left\{\frac{|M|M|J|}{C_g^2 H^2}\right\}_e\right)$$

$$\left. + \begin{bmatrix} -\delta_{e1} & 0 \\ 0 & \delta_{em} \end{bmatrix}\left(\left\{\frac{H^2}{2Fr^2}\right\}_e + \left\{\frac{M^2}{H}\right\}\right)_e \right) = \{0\} \tag{76}$$

These form the finite element algorithm for the quasi 1-D free surface flow equations. The standard matrix definitions are [1]

$$[A200] = \frac{1}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \qquad [A210] = \frac{1}{2|J|}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}$$

$$[A3010] = \frac{1}{6|J|}\begin{bmatrix} \begin{Bmatrix} -2 \\ -1 \end{Bmatrix} & \begin{Bmatrix} -1 \\ -2 \end{Bmatrix} \\ \begin{Bmatrix} 2 \\ 1 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 2 \end{Bmatrix} \end{bmatrix} \qquad [A3011] = \frac{1}{2|J|^2}\begin{bmatrix} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} & \begin{Bmatrix} -1 \\ -2 \end{Bmatrix} \\ \begin{Bmatrix} -1 \\ -1 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \end{bmatrix}$$

CHAPTER III

USER'S GUIDE TO *FATHOM*

This chapter details and explains the individual sub-programs of the FATHOM package. FATHOM consists of five separate programs originally written by G. S. Iannelli, cf. [2,6]. Two additional programs have been added to reduce the workload involved with running a simulation more than once. A general description of each sub-program is first presented, followed by a step-by-step guide to its use.

GENGRID - The Grid Generator Routine

1. General Description

GENGRID is an interactive program which builds data files for a grid suitable for a finite element flow analysis. It allows the user to readily create quite arbitrary grids constituted of quadrilateral elements. Input features include:

- Arbitrary number of linear, parabolic, or cubic boundary segments
- Fully automatic global node numbering
- Fully controllable uniform or non-uniform gridding from each segment of the boundary
- Immediate on-screen display of the grid

Upon GENGRID access, the user is prompted to respond to a series of simple questions about the desired grid. The information necessary to have in hand is illustrated in Figure 3, as constituted of:

- The number of columns and rows of nodes,
- The x and y coordinates of all the boundary segment end points, which may be chosen as linear, parabolic or cubic line segments. The boundary segment definition terminates where there is a change of boundary shape or gridding.

The center of Figure 3 is blanked out to emphasize that precise knowledge of location of the grid points is not necessary.

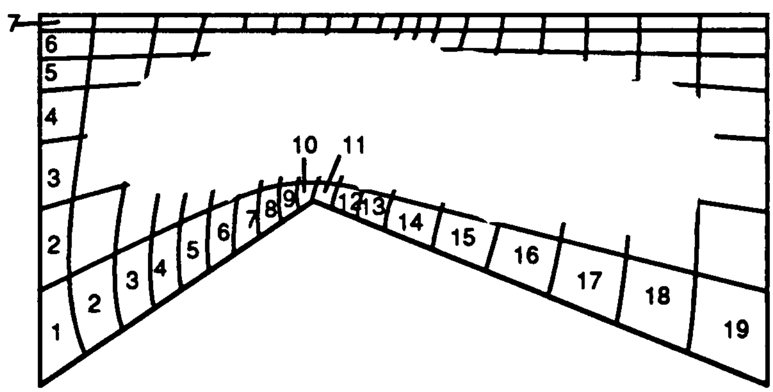

Figure 3. The necessary information for creating a grid.

The column and row indices are sufficient to locate any element. Also, the local node numbers, which are very simple to determine, are numbered 1 - 4 counter-clockwise beginning from the lower left corner of any element. Local node numbers allow specification of any particular node, if necessary.

The user is now prepared to generate the grid. A step-by-step tutorial of how to answer the questions posed by GENGRID is now detailed. One of the following computer terminals is selected:

- VT240 (or compatible terminal with graphics capability)
- Tektronics 4107, 4115 (or compatible terminal)

One of the following plotters is required if the grid is to be output as hard-copy.

- Tektronics 4662
- HP - 7550A

Several important guidelines for running a quasi one dimensional problem should be mentioned at the outset. The quasi one dimensional simulation requires two elements spanning the transverse direction. Hence, the grid must contain two straight rows of elements having equal width. The number of columns is arbitrary and may have non-uniform spacing if desired. Any width variations must be entered through the WIDTH subroutine in the interface program FEMFSI. Lastly, it is suggested (not required) that either the width or length of the grid be normalized to unity.

2. Tutorial

When GENGRID is accessed, the user (hereafter referred to as "you") is asked

```
DO YOU WISH TO GENERATE A NEW GRID OR VIEW A PREVIOUS GRID?
   0  >>  VIEW OLD GRID ONLY
   1  >>  CREATE A NEW GRID
```

If you choose to view an old grid only, you will be requested to select a device for the output and the particular type. The most recent grid generated will be then be shown as contained in the latest versions of the three data files:

FOR060.DAT, FOR062.DAT and FOR064.DAT.

If you choose to generate a new grid, you will be prompted

```
INPUT TOTAL NUMBER OF COLUMNS ?

      TOTAL NUMBER OF ROWS ?
```

For example, Figure 3. illustrates nineteen columns and seven rows, in which case the program responds

```
YOU HAVE         133 ELEMENTS
WHICH FORM       160 NODES ALTOGETHER
                  52 OF WHICH ARE BOUNDARY NODES.

Y CO-ORDINATE ZONES
TOTAL NUMBER OF ZONES?
```

There must be at least one zone specified. However, an additional zone one row tall is required if there is to be a cusp along the top or bottom boundary whose interior grid lines are to be smoothed. There must be a total of at least three zones specified in the *y* direction if cusps are to lie along both the top and bottom of the boundaries, one zone running along the top boundary, another zone along the lower boundary and the remaining interior rows comprising the final zone. Additional zones are also required for each interior grid line which has user-specified boundary departure angles (to be discussed shortly).

A *y* coordinate zone is a set of elements containing one or more complete rows. The shaded area in Figure 4 is a *y* coordinate zone of one row. This is necessary to ensure a smooth interior mesh. If there were no second zone specified, the boundary segment's sharp edge would propagate into the interior of the mesh. Thus, there are two *y* coordinate zones in Figure 4.

The next two requests are

```
ANY Y-BOUNDARY CUSPS ? (1=YES)
HOW MANY CUSPS ?
```

The program is actually requesting only the total number of cusps on the first and last columns of the boundary segment along which you desire smooth interior grid lines. If a grid is desired with interior cusps, then a negative (zero) response should be given. In the example, there are no cusps in the y direction.

The program then asks for the corresponding zone and cusp data in the x direction. These are entirely analogous to the y coordinate information except that the x coordinate zones run column-wise instead of row-wise. Thus the total number of cusps are along both the first and last rows. In the example, there is one zone in the x direction and there is one cusp in the x direction located along the first row at column 10. If there is an angle specified, it is the angle between the final grid line in the zone and the inward-pointing normal of the adjacent boundary segment (see Figure 4).

Note that there must be at least two y coordinate zones if there are any smoothed x boundary cusps. There must also be at least two x coordinate zones if there are any y boundary cusps. The zonal concept is an important feature to understand for the operation of the grid generator.

The next set of questions pertains to the boundary geometry of the grid.

```
    * BOUNDARY GEOMETRY *
HOW MANY BOUNDARY SECTIONS ?
```

Any change of boundary shape or the inclusion of boundary cusps requires a new section. For example, in Figure 4 there are five boundary section, all of which are straight lines. In addition, since boundary sections are where

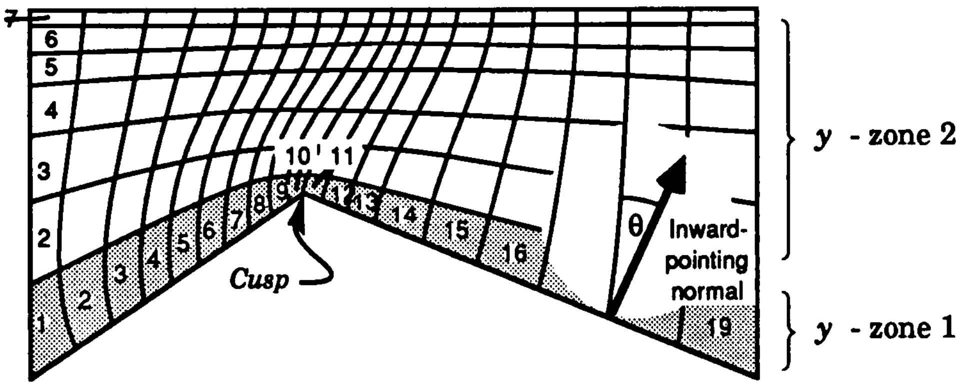

Figure 4. Grid specifications and the meaning of "specified angle" θ.

internal grid lines originate, any change of gridding must be done on separate boundary segments.

The location of cusps by row or column is then requested.

```
X CO-ORDINATE CUSP COLUMNS
CUSP1 COLUMN INDEX ?

WHICH ROW ?
1< FIRST ROW
2< LAST  ROW
```

X boundary cusps are requested by their column index, *y* boundary cusps by their row index. For example, in the problem above the cusp lies between column 10 and 11, thus the *x* boundary cusp is located in column 10 (it is equally permissible to specify column 11).

The program requires at least one *x* zone and one *y* zone, with an additional zone for each cusped boundary in either the *y* or *x* direction, respectively. If more than this number of zones is specified, it is assumed to be for the purpose of specifying the departure angle of the final grid line of the zone from its two end points.

```
Y-CO-ORDINATE-ZONE ROWS
ZONE              2

LAST-ROW INDEX ?

ZONE BOUNDARY-SHAPE SPECIFICATIONS?
SIDE              1
ORTHOGONAL          CODE: 1
ANGLE PRESCRIBED    CODE: 2


SIDE              2
ORTHOGONAL          CODE: 1
ANGLE PRESCRIBED    CODE: 2
```

The last row (or column, for x zones) to be included in each respective zone is requested along with its specified angle (Figure 4). The angle shown would be that of the x boundary zone which contained row 1, column 15. The angle is measured from the inward-pointing normal of the boundary counter-clockwise to the tangent of the grid line at the two boundaries.

The next sequence of questions under the topic of boundary segments is

```
FINAL NODE OF SECTION:              1
LOCATION BY ELEMENT CO-ORDINATES:
ELEMENT ROW ?

          1
ELEMENT COLUMN ?

         10
LOCAL NODE NUMBER ?

          2
ELEMENT              64                NODE             81
```

This sequence is repeated for each segment of the boundary. The global element number and node number on the bottom line are not input but echoed to the screen for verification of each section. For the example grid, the final node of the first segment is row 1, column 10, local node number 2. The beginning of segment number one as well as the end of segment five is assumed by the program to be the first row, first column, local node number one, with segments input sequentially running counter-clockwise. Again, a new segment is required any time a cusp, corner, change of boundary shape or change of grid is needed.

The x and y coordinates of the defined boundary section end points are the next input.

```
EXTREME-NODE CO-ORDINATES
FIRST NODE CO-ORDINATES
  X= ?
   0.0000000E+00
  Y= ?
   0.0000000E+00

SECTION            1
LAST NODE CO-ORDINATES
X= ?
   0.3333000E+00
Y= ?
   0.2000000E+00
```

The coordinates of the first node of the first element (i.e. the lower left corner node) are requested initially. The coordinates of the final node of every segment is then specified in order. Of course, the program assumes that the beginning of each segment corresponds to the end of the previous segment. For the example above, the beginning of segment one has coordinates (0.0, 0.0). The end of the first boundary segment has coordinates (0.33, 0.20). The end of the second segment has coordinates (1.0, 0.0), the third (1.0, 0.5). The end of the fourth segment has coordinates (0.0, 0.5). The end of the final segment is assumed to coincide with lower left corner node.

Boundary specifications are completed by giving the degree of boundary segment polynomial.

```
SECTION SHAPES

SECTION              1
X<CONST   Y<VAR           CODE: 1
X<VAR     Y<CONST         CODE: 2
X<VAR     Y<VAR , Y<IND   CODE: 3
X<VAR     Y<VAR , X<IND   CODE: 4
```

If either of the first two are chosen, the appropriate vertical or horizontal line is implemented. If either of the last two are chosen, then the program further asks the degree of polynomial desired

```
  I) LINEAR       Y   CODE: 1
 II) QUADRATIC    Y   CODE: 2
III) CUBIC        Y   CODE: 3
 IV) CUBIC SPLINE     CODE: 4
```

Segment one and two in the example grid both vary linearly, choosing x to be the independent variable. The third section is x held constant, y varying. The fourth segment is y held constant, x varying. The fifth segment is x held constant, y varying.

If the quadratic curve is chosen, the user is asked to specify the value of "$y(0)$." This is the value of the dependent variable at the midpoint of the boundary segment. If the cubic is chosen, the values of $y(^1/_3)$ and $y(^-{}^1/_3)$ are requested. These are the values of the dependent variable when the independent variable is located one third and two thirds the length of the segment, respectively, from the beginning of the boundary segment. Alternately, a cubic spline may be chosen as a boundary segment. The slopes of the end points are input in terms of the angle measured counter-clockwise from the x-axis. Due to large values of the tangent function, this angle cannot be close to 90°.

The next portion of the interactive grid generator queries as to the grid point distribution of each successive segment. The choices are

```
POINT-DISTRIBUTION
SECTION             1

GRID POINT UNIFORMITY:
UNIFORM      DISTRIBUTION    CODE: 1
LEFT  SIDE CONCENTRATION     CODE: 2
RIGHT SIDE CONCENTRATION     CODE: 3
BOTH  SIDE CONCENTRATION     CODE: 4
CENTER     CONCENTRATION     CODE: 5

STRETCHING STRENGTH  (1 THROUGH 20)
```

In the example, the grid points are concentrated on the right side for segment one, along the left side for segment two, on the right side for segment three, in the center for segment four, and on the left side for segment five. Reference to "left" or "right" is always made from a viewer outside the grid looking toward the interior of the domain.

The "stretching strength" is the intensity of the grid non-uniformity. A strength of 1.0 corresponds to uniform grid spacing, whereas a higher number yields a non-uniformly distributed grid. A small amount of experimentation is required to gain familiarity with the value appropriate to the task. FILEGRID (covered in the next section) is tailored to making such experimentation simple and rapid.

```
SELECT A DEVICE
1<< MONITOR
2<< PLOTTER
       1

SELECT A MONITOR
1<< VT220
2<< TEK 4114
3<< TEK 4207
```

The final request is the specific type of output device available. The grid is displayed scaled to the dimensions of the screen. The complete sample session of GENGRID is included in Appendix 5. It is very enlightening to run the example grid used above. If only the parameters stated above are utilized, a grid somewhat different from Figure 3, p. 34 is created. In order to correct this, a second x coordinate zone is added from columns 10 - 19. The orthogonal angle option is chosen on both the top and bottom. The resulting grid is as expected. Understanding this alteration is vital to the literate use of GENGRID.

FILEGRID - FOR055.DAT

FATHOM has incorporated a time-saving program, FILEGRID, allowing changes to be made to the grid quickly and simply. Every time GENGRID is run a log file entitled FOR055.DAT is generated. This file contains all of the input parameters for the grid. FILEGRID is an exact duplicate of GENGRID except for the input being from the FOR055.DAT file instead of from the terminal. You are asked whether you wish to build or simply view a grid. The only other user input is to select the type of terminal, as this may differ from time to time.

For example, after building a grid it may be determined that the exit boundary needs to be moved further downstream. Rather than having to repeat the entire procedure in order to build a new data file, FOR055.DAT may be edited, increasing only the two x coordinates of the exit boundary corners. FILEGRID may then be run in only a few seconds. Perhaps, after expanding the exit region that the grid is too sparse there. By changing the number of columns and the column coordinates of the exit plane corners in

FOR055.DAT, (for this simple problem, only three numbers) the grid may be quickly altered in just a few operations.

The output files are the same as for GENGRID

FOR060.DAT, the number of columns and rows, number of elements, nodes, boundary nodes, and number of nodes per element,

FOR062.DAT, the element nodal connection array and the boundary node vector array,

FOR064.DAT, the x and y coordinates of the nodes.

FEMFSI - The Data Preparation Routine

1. Overall Description

The data preparation routine, FEMFSI, is an interactive interface which builds the files necessary to run the FATHOM finite element analysis. The user must specify all initial and boundary conditions, problem parameters such as fluid properties, bed profiles and the Chezy coefficient, as well as the numerical constants such as artificial dissipation coefficients, time-step variation and time integration parameters.

The input data files required by FEMFSI are the three files generated by GENGRID: FOR060.DAT, FOR062.DAT and FOR064.DAT.

The output files generated by FEMFSI are

FOR059.DAT, the numerical dissipation (beta) parameters,

FOR061.DAT, the initial condition distribution,

FOR063.DAT, reference quantities,

FOR065.DAT, the initial inlet and outlet free surface height, the exit pressure and the inlet flow angle,

FOR066.DAT, the time integration parameters, FOR067.DAT, the current time, delta-time and time step variation parameters,

FOR068.DAT, information about each of the nodes (node codes), such as boundary condition(s) applied and number of variables per node,

FOR069.DAT, the bed slopes.

2. Tutorial

The session begins by requesting general information about the flow.

```
GRAVITATIONAL ACCELERATION ?

PLEASE SELECT ONE OF THE FOLLOWING

INITIAL-CONDITION PARAMETER CHOICES
1<< INLET Fr #  &  AVERAGE  DEPTH      SPECIFIED
2<< INLET Fr #  &  AVERAGE  VELOCITY  SPECIFIED
3<< INLET AVERAGE VELOCITY & DEPTH    SPECIFIED
4<< INLET Fr # & STAGNATION DEPTH     SPECIFIED
5<< INLET AVERAGE & STAGNATION DEPTHS SPECIFIED


DO YOU WISH TO SPECIFY THE REYNOLDS NUMBER?
1 << YES
2 << NO

        1

REYNOLDS NUMBER SPECIFICATIONS
SELECT ONE OF THE FOLLOWING COMBINATIONS

1 << REFERENCE VELOCITY
     REFERENCE LENGTH AND
     EDDY VISCOSITY  SPECIFIED

2 << REFERENCE LENGTH
     REYNOLDS NUMBER AND
     EDDY VISCOSITY  SPECIFIED

...

GENERALIZED CHEZY COEFFICIENT ?

DO YOU PREFER A QUASI 1-D ANALYSIS ?
1<< YES
2<<  NO
```

```
ARE YOU STUDYING A TRAINING-STRUCTURE
   PROBLEM ?
   1<< YES
   2<<  NO
```

The "initial condition parameter choices" questions is followed by an inquiry of the chosen variables. A consistent set of units must be adhered to throughout. If the Reynolds number is specified, the data are input at the "..." above. Otherwise, the reference values of velocity and length are assumed equal to the inlet velocity and width respectively, while the kinematic viscosity is assumed to be that of sea water at 68° F in units of ($ft^2/_s$). Although the actual training structure data must be included by running the TSGEN program, a positive response to this question will set a flag in the FOR063.DAT file, appropriately directing subsequent data processing.

If the problem involves training structures, the program at this point assists the user in specifying an "auxiliary channel." The auxiliary channel allows you to specify flow that initially circumvents any training structures present. This takes place via the specification of the x and y coordinates of each of the four nodes defining a quadrilateral region surrounding a training structure, see Figure 5. The shape of each "wedge" is at the user's discretion and the complete auxiliary channel is formed by joining these wedges as indicated in Figure 6.

The next program section sets up the initial conditions. The previous version of FOR061.DAT is used as an initial condition if none is created. The initial conditions are created as follows:

```
DO YOU WISH TO SET UP INITIAL CONDITIONS ?
1<< YES
2<<  NO
```

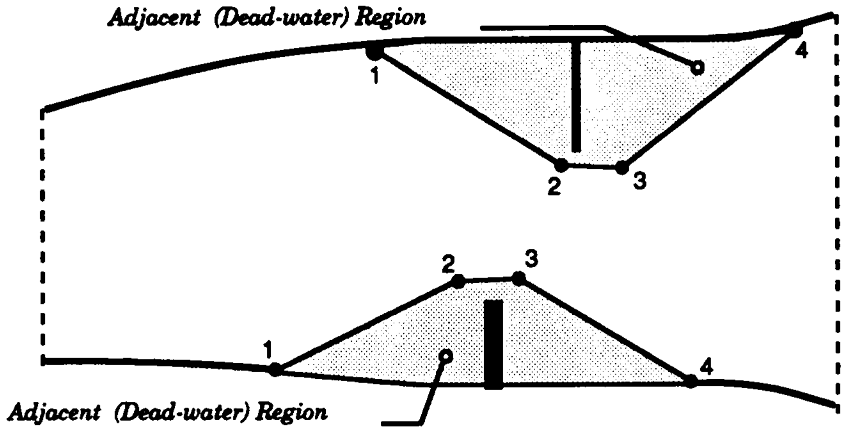

Figure 5. Trapezoidal wedges surrounding training structures.

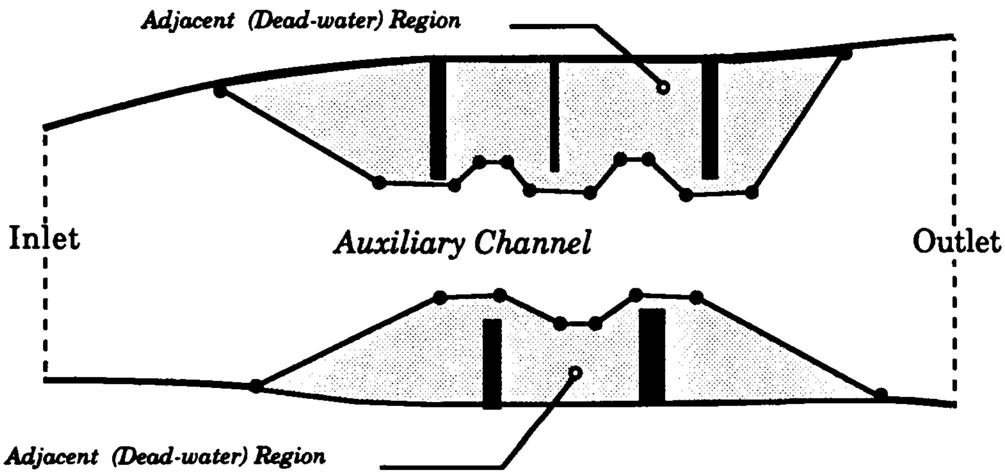

Figure 6. An assembly of trapezoidal wedges.

```
DO YOU WISH TO SPECIFY AN AVERAGE INITIAL OUTFLOW HEIGHT ?
1<< YES
2<<  NO

IS THE INITIAL DEPTH DISTRIBUTION KNOWN ?
1<< YES
2<<  NO
```

The stated initial outflow height must be uniform across the outlet. If the initial depth distribution is known, it must be either directly imbedded in the HDISTR subroutine or supplied via longitudinal troughs (Appendix A).

```
PLEASE SELECT THE TYPE OF FLOW
1<< INVISCID FLOW WITH CONSTANT U ALONG Y
2<< VISCOUS  FLOW
3<< INVISCID FLOW WITH VARIABLE U ALONG Y

DO YOU WISH TO UTILIZE THE DEPTH-SPECIFICATION ROUTINE ?
1<< YES
2<<  NO

OUTFLOW HEIGHT CONSTRAINED ?
1<< YES
2<<  NO

INLET FLOW DIRECTION PRESCRIBED ?
1<< YES
2<<  NO
```

The three choices of velocity profiles across the channel permit initial conditions for a wide variety of flows. The inviscid flow with constant velocity across the channel is suitable for flows with no training structures present. The viscous flow (parabolic profile) option is appropriate for any flow having zero velocity along the side boundaries. If uniform flow developing into a parabolic profile (for inviscid problems with spur dikes) is desired, two options are offered. The flow may develop into either a full parabolic (no slip on either the top or bottom) or a semi-parabolic profile (having slip flow on either the top or bottom).

Constraining outflow pressure fixes the average hydrostatic pressure of the fluid at the outlet boundary to its initial value. Imposing outflow pressure rather than height gives the advantage of allowing both the fluid height and velocity to vary across the exit boundary. Prescribing the inlet direction dictates the flow direction throughout the domain. The inlet flow direction is input in degrees measured counter-clockwise from the x-axis. Since initial conditions are basically quasi one-dimensional, only one inlet flow direction is allowed (although more than one inlet could be defined).

The time integration parameters deal with the solution of the ordinary differential equation resulting from the finite element assembly. The previous values are used if none are created. These are saved in the file FOR066.DAT.

```
DO YOU WISH TO SET UP TIME-INTEGRATION
ALGORITHM PARAMETERS ?
1<< YES
2<<  NO

TYPE OF INTEGRATION ALGORITHM ?
1=EULER BACKWARD
2=CRANK-NICOLSON
3=IMPLICIT RUNGE-KUTTA

DO YOU WISH TO SET UP TIME-STEP
VARIATION PARAMETERS ?
1<< YES
2<<  NO

INITIAL TIME ?

TIME MAX ?

TOTAL NUMBER OF STEPS ?

MINIMUM TIME STEP ? (< INITIAL DT !)

INITIAL TIME STEP ?

MAXIMUM TIME STEP ?

USER DEFINED ZERO ?

DT-VARIATION PARAMETER (0.0< PAR <=1.0)
```

The user-defined zero is the value of the function which represents sufficient convergence during each time step. Specifically, during the Newton's method, the root of the function is sought. The actual value obtained is generally never exactly zero, so this "epsilon" value is sufficient to indicate that a root has been found. The time-step variation parameter controls the size of the maximum variation allowed for the flow variables. A larger value may permit more frequent time step increases, if convergence is obtained. The previous values are used if none are created. These are saved in the file FOR067.DAT.

Artificial dissipation mechanism coefficients are requested next. The coefficients, β_1, β_2 and β_3 control the numerical dissipation level for fluid height, h, the x component of momentum m_x, and the y component of momentum m_y, respectively. This mechanism is available to exercise control over the underlying numerical dispersion error mechanisms if necessary, as discussed in Chapter 1.

The bed data may either be entered as elevations or as the bed slopes. This is accomplished through the function BEDEL or the subroutine BEDSL respectively. These values may be input either by means of an explicit function or a series of IF...THEN statements containing local elevations or slopes.

```
DO YOU WISH TO SET-UP CHANNEL-BED DATA ?
   1<< YES
   2<<  NO

   TYPE OF CHANNEL-BED INPUT DATA

   1<< BED ELEVATION AT MESH NODES
   2<< BED SLOPES    AT MESH NODES
```

The boundary condition information is input by boundary section. The sections referred to here, however, are not necessarily the same as the segments used in the grid generator. The boundary is decomposed only to the degree such that each section has only a uniform boundary condition specified. For the grid generation problem illustrated before (see Figure 7) having flow from left to right, the eight boundary condition sections are

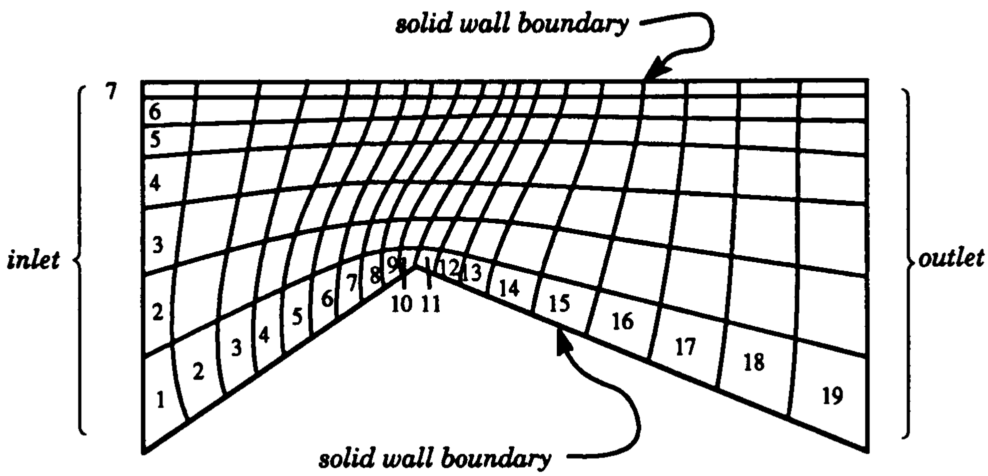

Figure 7. A typical grid upon which boundary conditions are to be imposed.

- inlet from row 1 to row 7 along column 1
- inflow boundary corner at row 1, column 1, local node number 1
- wall boundary from column 1 to column 19, along row 1
- outflow corner at row 1, column 19, local node number 2
- outlet boundary from row 1 to row 7, along column 19
- outflow boundary corner at row 7, column 19 local node 3
- wall boundary from column 19 to column 1, along row 7
- inflow boundary corner at row 7, column 1 local node 4.

This information is requested in the following manner beginning from the lower left hand corner of the grid and continuing counter-clockwise.

```
HOW MANY BOUNDARY SECTIONS ? (<=20)
          8
          8 SECTIONS
NOW ENTER:  C ,TO CONFIRM
C

SECTION              1
 TYPE OF BOUNDARY ?
1<< INFLOW   BOUNDARY
2<< INFLOW   BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL     BOUNDARY
     2
THE TYPE SELECTED IS            2
NOW ENTER:  C ,TO CONFIRM
C

SECTION              1
TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   h  & du/dn=0 & dv/dn=0
3<<   mx & dv/dn=0
4<<   my & du/dn=0
5<<   h  & mx & dv/dn=0
6<<   h  & my & du/dn=0
7<<   mx & my
8<<   h  & mx & my
9<<   u  & dv/dn=0
10<<  du/dn=0 & v
11<<  u  & v
12<<  mx & v
13<<  u  & my
14<<  h & my/mx
15<<  my/mx
          8

THE TYPE SELECTED IS
NOW ENTER:  C ,TO CONFIRM
        C
```

For any outflow boundary section, you are asked whether outflow pressure is to be fixed (assuming neither outflow height or pressure has been previously constrained). For any outlet boundary, an additional question asked is

```
DOES THIS OUTLET FACE THE AXIAL DIRECTION?
```

A flag is set directing the main processor to correctly handle the corner boundary condition. This option allows outlets in the cross-stream direction. Axial is defined to be in the direction of the inlet velocity.

The two points defining the extremities of each boundary section are requested.

```
FIRST
EXTREME NODE OF SECTION            1
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          1
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
          1
ELEMENT            1 NODE            1
LOCAL NODE NUMBER:            1
ROW            1 COLUMN            1
NOW ENTER:  C ,TO CONFIRM
        C

SECOND
EXTREME NODE OF SECTION            1
NODE LOCATION
ELEMENT CO-ORDINATES

ELEMENT ROW ?
          1
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
          1
ELEMENT            1 NODE            1
LOCAL NODE NUMBER:            1
ROW            1 COLUMN            1
NOW ENTER:  C ,TO CONFIRM
        C
```

FEMFILE - FOR054.DAT

FEMFILE is a time saving program allowing changes to the problem parameters quickly and straight-forwardly. Each time FEMFSI is run, a log file entitled FOR054.DAT is generated. This file contains all of the keystrokes input during the FEMFSI session. FEMFILE is an exact duplicate of FEMFSI with the exception of the input being from the FOR054.DAT file instead of from the terminal. Instead of having to respecify everything, the user may simply edit file FOR054.DAT to alter any changes to the problem. This can save much time and effort when the changes to the problem (especially the boundary conditions) are minor.

For example, after running a problem it may be determined that a pressure boundary condition would be more appropriate at the exit than a fixed value of h. Rather than having to repeat the entire procedure in order to build a new data file, FOR054.DAT may be edited changing only a few values. Then the program FEMFILE may be run in just a few seconds to build the data files with only the desired changes.

TSGEN - The Training Structure Routine

TSGEN is the training structure generator. It automatically inserts the proper boundary conditions necessary to simulate channel flow obstructions. To utilize the training structure generator, GENGRID is first run as if there were no training structures in the flow. TSGEN is run after FEMFSI and prior to running FEMFRS.

Either right triangular or rectangular training structures of any proportion may be inserted. TSGEN is the simplest of the interactive programs to use. The first request is for the total number of training structures desired following by the element coordinates of their lower left-most node and of their upper right-most elements. The choice of a viscous or inviscid simulation is required to correctly handle the boundary node codes. The lower left and upper right most element of the rectangular training structure are entered as the initial and final elements. For a triangular training structure, the lower left-most element and the upper right-most element of the rectangle which contains the right triangular training structure are entered. These specified elements form the boundary of and are internal to the training structure.

The "rectangular" and "triangular" training structures are rectangular or triangular only if the grid is rectilinear in the region of the training structure. The training structures follow element boundaries and thus are curved if the included elements have curved sides.

If a training structure is to be utilized on an inlet or an outlet, the node codes should be verified in the FOR068.DAT file. A code of seven indicates the the node is internal to the training structure. Training structures are handled in the code by first neglecting all contributions from the interior nodes and then setting their dependent variables to nought.

A rectangular training structure must be at least three nodes wide and three nodes high. Further, it must not come within three nodes of a boundary unless it is connected to that boundary (see Figure 8). The solid black nodes shown are boundary nodes and hence active in the numerical approximation. The white nodes are active nodes internal to the flow domain. The nodes shown in hatching are internal to the training structure and are hence reduced out of the finite element matrix statement.

For triangular training structures, the situation is slightly more complicated. The basic concept is similar, however, the grid must be altered to establish a smooth wall over the inclined surface of the dike (Figure 9.).

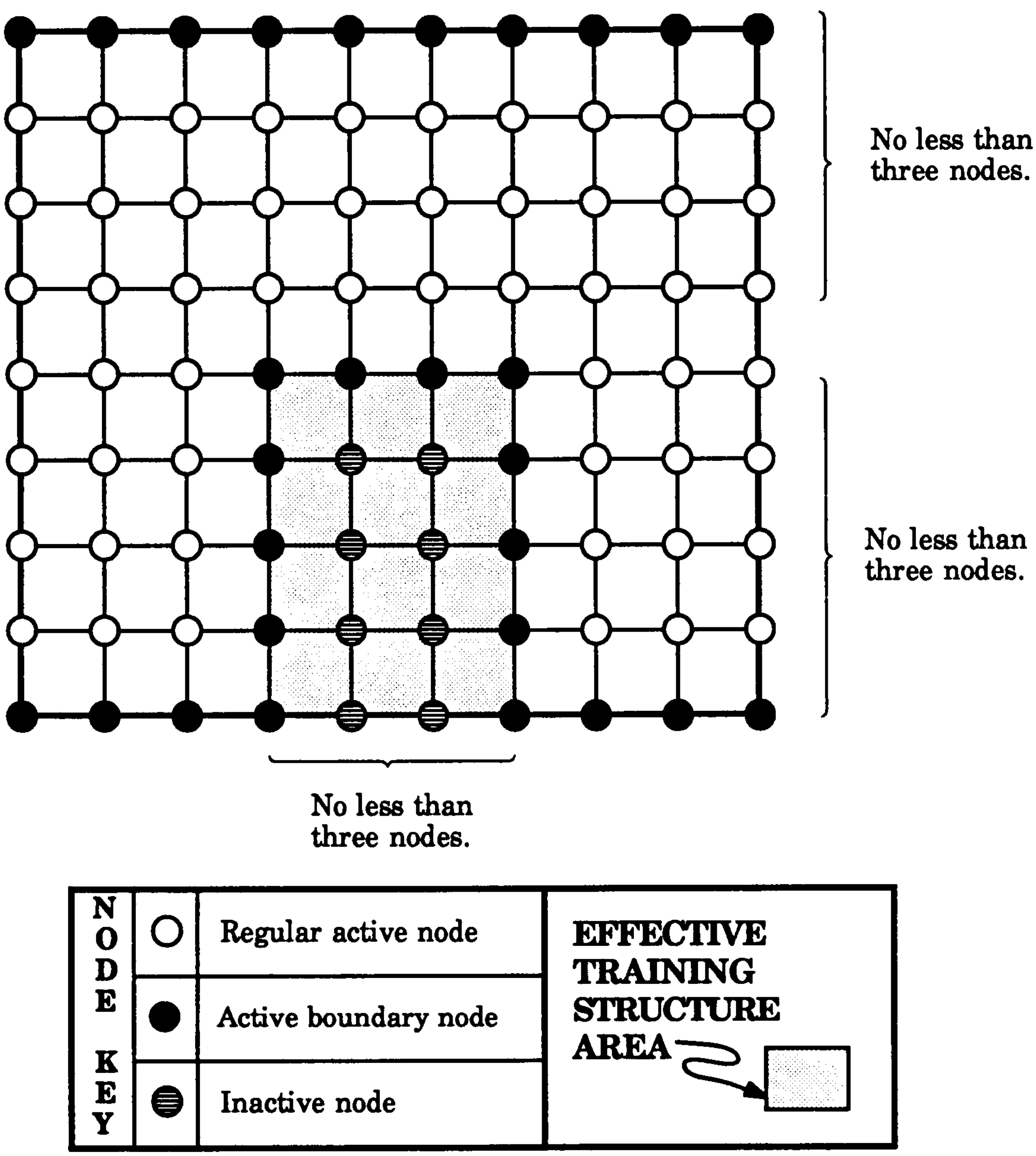

Figure 8. Rectangular training structure diagram.

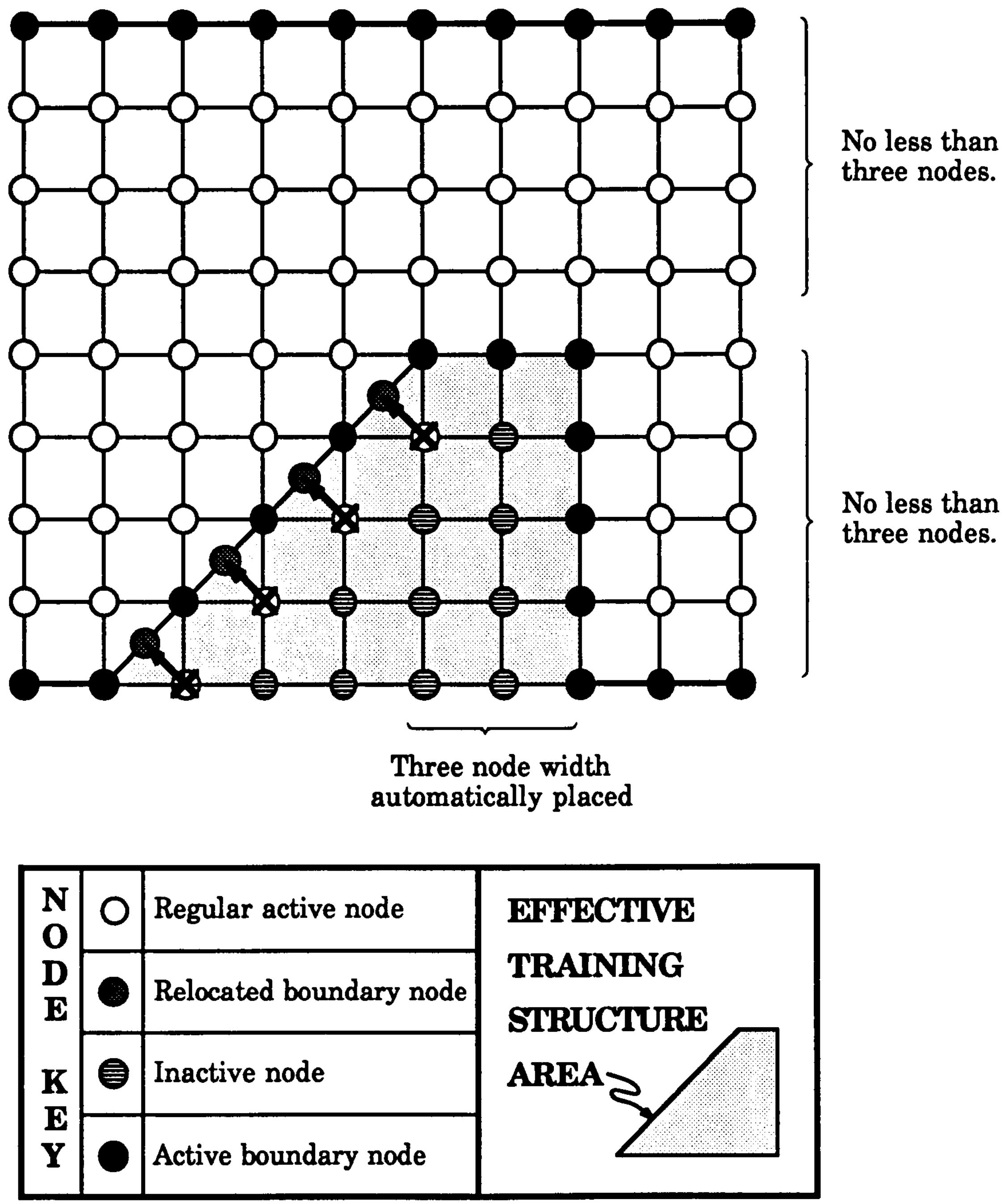

Figure 9. Triangular training structure organization.

FEMFRS - The Finite Element Solving Routine

FEMFRS is the main finite element algorithm. All necessary data files have been prepared by the previous programs. The program is normally executed in batch mode with all necessary data files accessible. The output files of this program are FOR070.DAT the time parameter update, FOR071.DAT, the solution array, and FOR072.DAT.

FOR072.DAT contains an integer map of the computational coordinates, and initial condition information, bed data and the solution of all variables as well as boundary conditions, flow parameters, time integration information, the residuals, norms and temporal information at each time step. FOR071.DAT contains the solution of the problem variables. This file must be renamed FOR061.DAT (the initial conditions) for any further runs of the same problem. In addition, FOR061.DAT is the file which is used to produce the output of JOEGRAPH, the solution graphics program.

JOEGRAPH - The Solution Graphics Routine

1. General Description

JOEGRAPH (so-named after its author Joe Iannelli) is a DISSPLA-based graphics routine to generate single or multiple contours plots, surface plots, velocity vector plots, or any of these projected onto a plane. The option exists for viewing these graphs from any orientation.

For contour plots, the program determines the maximum and minimum value of the function to be plotted. Knowing the number of contour levels desired, the function value at each level is calculated. The routine tracks contours along the mesh in the following manner. The value of the contour level is first computed. Each finite element is examined to determine whether or not the current contour intersects its boundary. If no intersection is found, the following element is examined until an element is identified which intersects the current contour.

At this point, the contour tracking begins. The contour is traced through the finite element by interpolation. The coordinates of the element into which the contour enters (adjacent to the current element) are determined. If the element from which the initial tracking began is reached, then a closed contour is assumed and the tracking is halted. If a boundary is reached, then the tracking direction is reversed until another boundary is reached. Once every element has been checked, the next contour level is determined and the procedure repeated. Each contour interval may be assigned either a specific color or black and white hatching pattern.

For the surface plots, the function data themselves are used as coordinates in the z-direction. The solution surface represents a rigid body in space which may be revolved and projected onto a viewing plane by a transformation of coordinates. This transformation is accomplished using the Euler angles. The coordinates of the rigid body plot are then mapped onto a two-dimensional viewing plane.

Velocity vector plots are obtained by plotting an arrow in the direction of and proportional to the length of the local velocity vector. The magnitude of the velocity may also be represented by a color, if desired.

The same equipment is required as was for the GENGRID program. The additional choice of an IMAGEN printer is available for high resolution black and white output. Unless the IMAGEN option is chosen, JOEGRAPH generates no output file and all data is sent to the terminal. However, it requires the input files:

FOR060.DAT - the number of columns and rows, number of elements, nodes, boundary nodes, and number of nodes per element (four), FOR061.DAT - the solution field data

FOR062.DAT - the element nodal connection array and the boundary node vector array,

FOR064.DAT, the x and y coordinates of the nodes.

FOR068.DAT - the node codes

2. Tutorial

You are first requested

```
 PROCEDURES AVAILABLE:

1<< SURFACE PROJECTED ON A PLANE (MESH)
2<< SURFACE
3<< VELOCITY FIELD PROJECTED ON A PLANE
4<< VELOCITY FIELD ON SURFACE
5<< CONTOUR PLOT PROJECTED ON A PLANE
6<< CONTOUR PLOT PROJECTED ON A PLANE WITH
        AREA SHADING
7<< CONTOUR PLOT ON SURFACE
8<< CONTOUR PLOT ON SURFACE WITH
        AREA SHADING

PLEASE ENTER THE NUMBER OF PROCEDURES YOU
WISH TO HAVE CARRIED OUT SIMULTANEOUSLY.

PLEASE ENTER THE NUMBER OF
HEADING LINES (L<=4)
   1

PLEASE ENTER HEADING LINE #            1
   Joegraph Plot

PLEASE ENTER THE VERTICAL DISTANCE
(IN INCHES) BETWEEN PLOTS.(>0)
```

Input the number of graphs to plot on the page. The first graph will be located on the bottom, the second next to the bottom, and so forth. The vertical space between the plots is asked for next.

Eight types of graphs are available

```
1<< SURFACE PROJECTED ON A PLANE (MESH)
2<< SURFACE
3<< VELOCITY FIELD PROJECTED ON A PLANE
4<< VELOCITY FIELD ON SURFACE
5<< CONTOUR PLOT PROJECTED ON A PLANE
6<< CONTOUR PLOT PROJECTED ON A PLANE WITH
       AREA SHADING
7<< CONTOUR PLOT ON SURFACE
8<< CONTOUR PLOT ON SURFACE WITH
       AREA SHADING

PLEASE ENTER THE CODE FOR TASK NUMBER:            1

    2

PLEASE ENTER THE INITIAL OF THIS
TASK'S INDEPENDENT VARIABLE.
```

Depending upon which task is chosen, you will be prompted to answer the following:

- Choice 1:

  There are no questions asked for task 1.

- Choice 2:

  You are asked which variable you desire to plot *h, u,* or *v,*

- Choice 3:

  You are asked if you want color vectors and if you want to specify custom vector viewing parameters. If so, then you are asked:

```
ENTER VELOCITY-VECTOR MAGNIFICATION
CONSTANT

ENTER VECTOR WIDTH-TO-LENGTH  0-5

ENTER VECTOR SIZE 0-6

ENTER COLUMN STRIDE

ENTER ROW STRIDE
```

- vector magnification scales the length of the arrows; default= 1,
- vector width to length ratio determines thickness of the arrowhead used to represent the vector; default = 0,
- vector or arrowhead size, 0 is the smallest, 6 the largest; default=1,
- column stride controls the number of nodes to skip between each arrow plotted in a given column, i.e., the number added to the row index to obtain the next location for displaying an arrow. If the column stride is two, every other node in a given column gets an arrow; default: 1,
- the row stride controls the number of nodes to skip between each arrow plotted in a row,

- Choice 4:

  After being asked if you prefer color vectors and a color scale, you will also be asked which variable the surface will represent *h, u,* or *v*. You will then be asked whether or not you wish to specify custom vector viewing parameters. If so, you will be asked the same set of questions listed for choice three above.

- Choice 5:

  You are asked which variable to plot *h, u,* or *v*. and whether you want a separate color scale plotted for the contours.

- Choice 6, 7 or 8:

  You are asked which variable the surface will represent, *h, u,* or *v*.

If either task 6 or 8 is chosen, you will be asked whether you would like color contours or 'black-and-white' hatching. This option is given so that contours may be differentiated on monotone terminals or plotters. For each contour plot you request, you are prompted

```
FOR CONTOUR PLOT NUMBER:           1
MIN=   -1.331216     MAX=    4.564607

DO YOU WISH TO SPECIFY THE
CONTOUR LEVELS BY DESIGNATING
THEIR UPPER AND LOWER LIMITS,
ALONG WITH THE # OF CONTOURS ?
0 <<  NO
1 << YES
1

ENTER ZMIN
0.

ENTER ZMAX
1.

ENTER # OF LEVELS (L>=2)
18
```

If you wish to manually assign contour level values, enter upper and lower levels with the number of contours between these values. The upper and lower values need not be in the range of minimum and maximum values of the plotted variable. For example, you could choose a range of zero to one and ten levels. This would specify a level at each 0.1 contour of the function.

You are then asked if you would like custom viewing parameters. This section allows the user a completely arbitrary viewing location of the results. If you accept the default values instead, graphics will be viewed from directly above the horizontal plane of the mesh. This is appropriate for vector and contour plots, but not for surface plots. If custom parameters are chosen, you will be asked the following questions:

```
ANGLES IN DEGREES
ENTER 1-ST ESCAPE ANGLE :
ETA  =

ENTER 2-ND ESCAPE ANGLE :
SIGMA  =

ENTER EULER ANGLES :
PHI =
```

```
THETA=
PSI=

ENTER REFERENCE POINT COORDINATES:
XREF=

YREF=

ZREF=
```

Escape angles are the angles of the coordinate axes with respect to the screen horizontal, measured counter-clockwise, see Figure 10. The Euler angles are the degrees of rotation about the z, x_1, and z_2 axis in that order, where the subscripts refer to subsequent positions of the axes. For example, z_2 is the position of the z axis after two rotations have occurred. Since finite rotations are not additive vectorially, the results are not generally intuitively obvious *a priori*. Hence, some amount of trial and error will normally be required for all but simple cases. It is helpful in predetermining the outcome to limit these to one or at most two non-zero rotations. Although somewhat cumbersome, the generality of these rotational angles allow one to view the solution from any vantage point whatsoever.

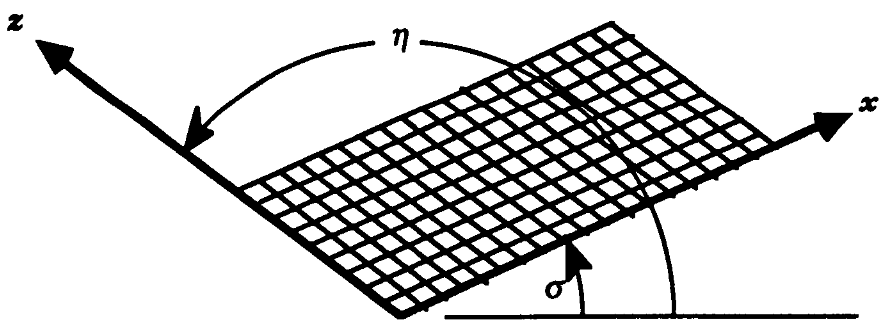

Figure 10. Definition of sigma and eta, the escape angles.

Reference point coordinates (XREF, YREF, ZREF) offset the image on the screen by selecting the origin. The next inquiry is

```
PLEASE ENTER X-AXIS MAGNIFICATION CONSTANT
PLEASE ENTER Y-AXIS MAGNIFICATION CONSTANT
```

The x and y magnification factors allow different scaling of the two axes. This is helpful for two reasons. In the case of a very narrow channel, the output appears much more balanced and pleasing if the short axis is increased in scale. It also helps resolve detail in geometries having a high element density in one direction.

The next questions request the type of output desired. This inquiry is similar to that of the GENGRID routine, page 43-44, except that an additional choice of an IMAGEN printer is given. If selected, this will create a formatted output file entitled IMAGEN.IMP. All dimensions should be specified in inches. The text style and size parameters are also controlled in this page set up inquiry. If you set up a custom page, then the following questions are asked:

```
ENTER CHARACTER HEIGHT
ENTER HEADING HEIGHT FACTOR
ENTER PAGE HEIGHT
ENTER PAGE WIDTH
AREA HEIGHT
AREA WIDTH
ENTER MAGNIFICATION FACTOR
```

- Character height, in inches, of the text. Default value is 0.14 inches.
- Heading height factor is the ratio of the the header to other characters on the screen. Hence entering 1.0 would give uniform text size. Entering 2.0 would make the heading twice as large as the other characters. Default is 2.0.
- Page height is the vertical dimension in inches of the video screen or plotter page.
- Page width is the dimension of the the screen or page width in inches.
- Area height and area width are the dimensions of the image area of the screen or page. If a one inch margin is needed on all sides of an 8.5" x 11" page, then the image area must be 6.5" height and 9" width.
- Magnification factor enlarges or reduces the final graph. For example, 0.5 is half scale, 1.0 is full size, 2.0 is double scale, etc. This is helpful for viewing specific areas of the flow in greater detail. Default is 1.

This is the end of the custom page parameter section.

Finally, for any plot containing a surface, you are asked,

```
TASK NUMBER              1
DO YOU WISH TO STRETCH
THE CONTOUR PLOT TO SPECIFIED
MINIMUIM AND MAXIMUM VALUES ?

0 << NO
1 << YES
```

This feature is handy for viewing the details of a "flat" solution, i.e., one having only small variations in the function values. This has no effect on the contour levels, only the "*z*" scaling is affected by this feature.

CHAPTER IV

RESULTS AND DISCUSSION

Six example problems, as solved using the FATHOM finite element procedure, are now investigated. These cover geometries from a simple straight channel to a channel having ten spur dikes and being highly non-uniform in cross section and bed depth. The flow geometries are:

1. a straight-walled, uniform-width channel having a sharp transverse rise in bed elevation.
2. a converging/diverging channel producing a transition from sub-critical to super-critical flow and return.
3. a channel with a single slender rectangular spur dike extending approximately one half channel width into the flow.
4. a channel containing a single triangular spur dike extending approximately one half channel width into the flow.
5. a meandering channel with a horizontal bed.
6. a simulation of the Mississippi River outlet into the Gulf of Mexico containing ten spur dikes modelling the actual river geometry and bed topography.

The description of each of these problems is detailed regarding specifications, comparison to theory or expected results and comments on code execution and features.

The Bed-Step Problem

The straight, uniform channel having a sudden increase in the bed elevation at one third channel length as illustrated in Figure 11, possesses solutions which are directly comparable to theory. Since no geometrical variations occur in the transverse direction, the problem is quasi one dimensional. Two cases are investigated. The first utilizes a sub-critical inlet, Froude number of 0.3 and the second a super-critical inlet, Froude number of 2.0. The inlet height is fixed at one meter for the sub-critical case and 0.28231 meters for the super - critical case (This yields the same critical free surface height as the first case). The transverse bed-step is 0.2 meters high, and the channel length is ten meters.

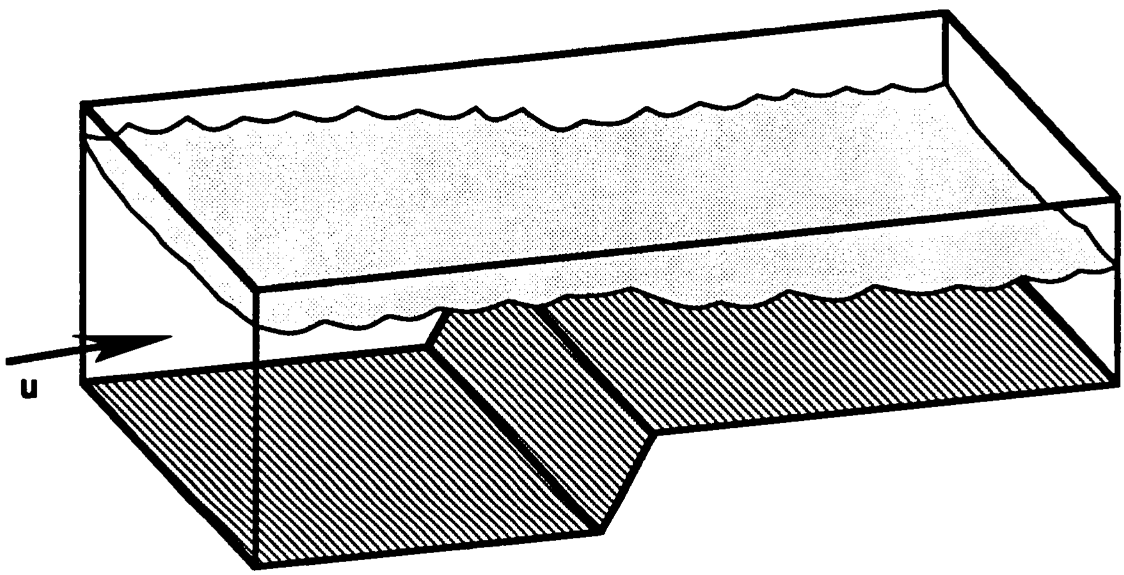

Figure 11. Geometry for the quasi-1-D transverse bed step problem.

Initial conditions at each node are generated by using the known uniform width and inlet Froude number. Thus a uniform velocity profile and constant depth are calculated initially. The non-uniform computational mesh, Figure

12, contains twenty axial elements with clustering about the bed-step. The coefficients β_1, β_2 and β_3 are set to (0.3, 0.3, 0.0). Only the first two are utilized in a quasi-one-dimensional simulation.

Due to the poor approximation of the initial conditions to the steady solution, the sub-critical simulation undergoes several large axial surges before reaching steady state. This experience prompted a new initial condition generation scheme for flows with varying bed profiles. The super-critical flow converges very rapidly, even with the poor initial approximation. Since disturbances cannot travel upstream in a super-critical flow, the incorrect initial condition is quickly simply swept out of the computational domain.

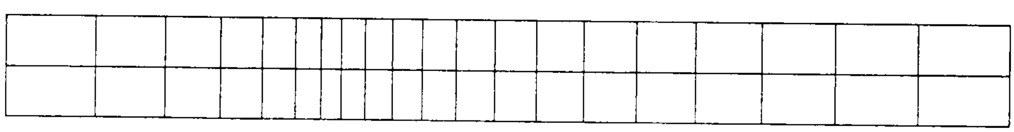

Figure 12. The quasi one-dimensional bed step problem grid.

The final numerical solutions vary from the analytical solution by no more than 1.4%. This may be reduced further by employing a denser mesh in the vicinity of the jump, since resolution of the discontinuity requires approximately two nodes. The resulting distributions of depth and velocity for the two flows are shown in Figure 13. Notice the effect of the bed step is to increase the speed and reduce the height of a sub-critical flow. Conversely, the effect is to slow down a super-critical flow, and increase its depth. In both cases, a positive bed slope causes the flow to approach the critical state.

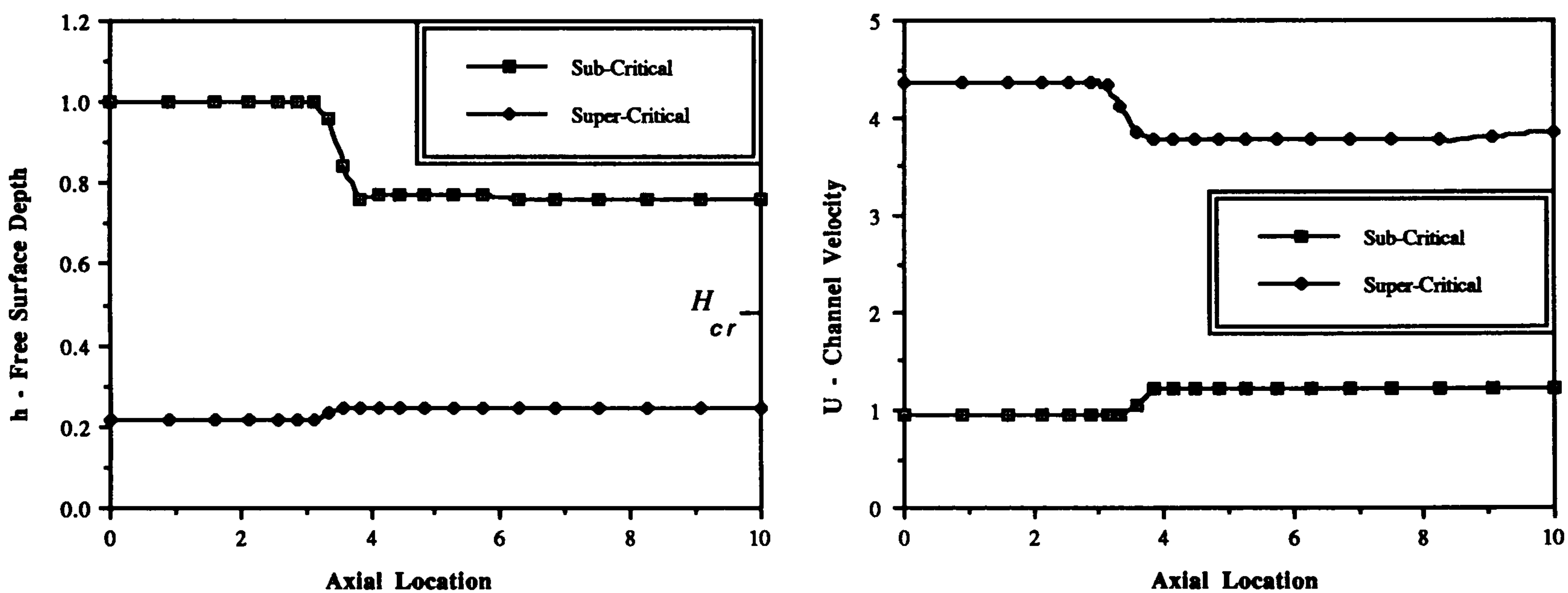

Figure 13. The quasi one-dimensional bed step problem solution.

The Super-Critical Nozzle Problem

The second problem is a converging/diverging channel of uniform bed elevation capable of producing a super-critical to sub-critical discontinuity, i.e. a hydraulic jump, in the exit region. All pertinent flow regimes, as characterized by Froude number, are investigated. Initially the sub-critical inlet flow, $Fr = 0.226$, accelerates to almost critical $Fr = 0.97$ at the throat then reexpands to a sub-critical downstream flow. The exit pressure is lowered slightly from this initial condition. The outlet free surface depression travels upstream until reaching the throat. The flow at the throat becomes critical and on the ensuing expansion to super-critical flow begins to travel downstream. A hydraulic jump is produced, separating the super-critical and sub-critical flow regions beyond the throat. Since the exit height was reduced only slightly, the jump propagates to a stationary (steady state) point well within the computational domain. The exit height is further reduced until the hydraulic jump becomes flushed out of the channel dependent upon the exit height. The effect of adjusting various parameters is investigated in order to better understand their behavior on this class of flows.

The problem is treated as quasi one dimensional with a uniform grid employing 100 element columns and two element rows. However, the width is distributed sinusoidally (via specification in the subroutine WIDTH in the user interface program) such that the throat area equals 40% of the inlet width. The width increases again beyond the throat until it reaches the exit, where the cross section equals the inlet value. The inlet height is specified to be 1.0 meter. The Chezy coefficient is initially set to 1.E+08 (i.e., no bed drag), and later adjusted to verify its dissipative effects on the flow. The beta parameter set initially is set to {0.3, 0.3, 0.0}. This, too, is adjusted to seek a better behaved solution. The flow was assumed psuedo-turbulent with constant eddy viscosity such the effective Reynolds number based on the inlet width is 3500.

Figure 14 shows the initial condition Froude number distribution. Figure 15 shows the Froude number after the free surface height reduction at the exit has propagated some distance upstream. Figures 16 - 17 show the Froude number and x-momentum distributions after the disturbance has travelled from the throat to the steady state hydraulic jump located in the diverging section of the channel. In accordance with the characteristics analysis of Chapter 2, the boundary conditions were altered while running the test cases. For sub-critical flows, one variable must be constrained at both the inlet and outlet, whereas for super-critical flows, two variables are fixed at the inlet and the outlet is unconstrained.

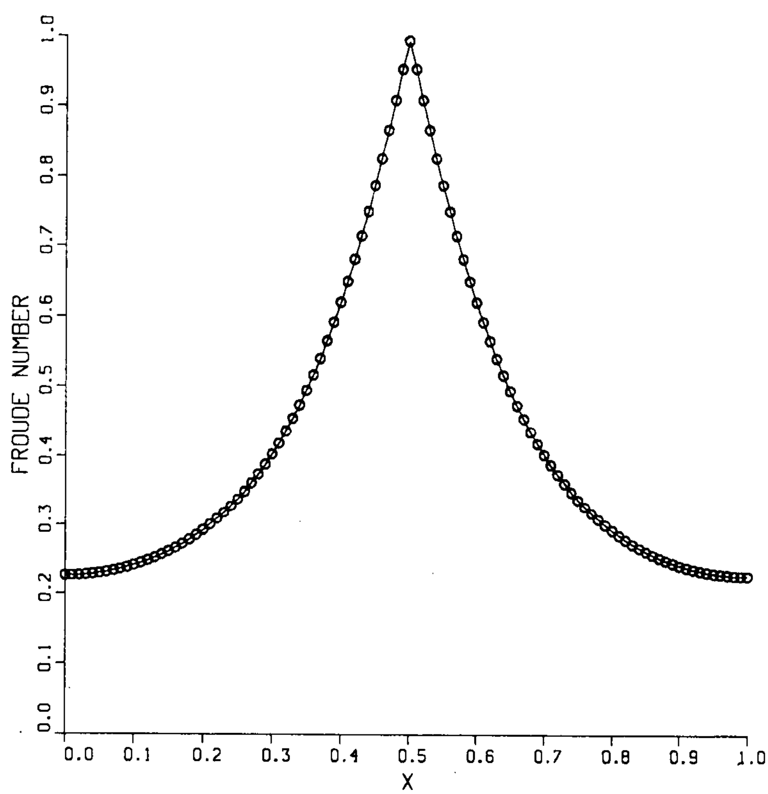

Figure 14. Quasi 1 - D bed step - initial Froude number distribution.

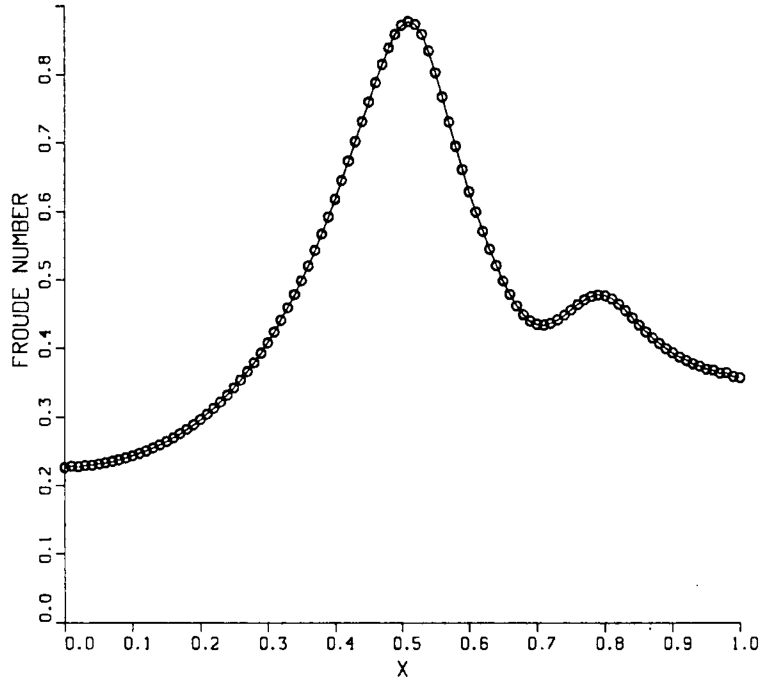

Figure 15. Froude number variation after reducing exit pressure.

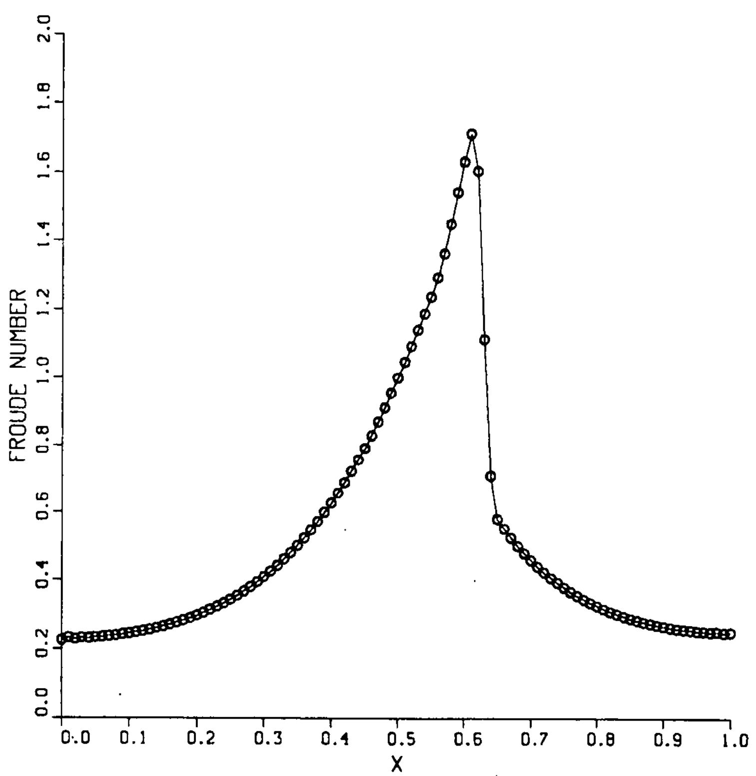

Figure 16. Steady state Froude number distribution.

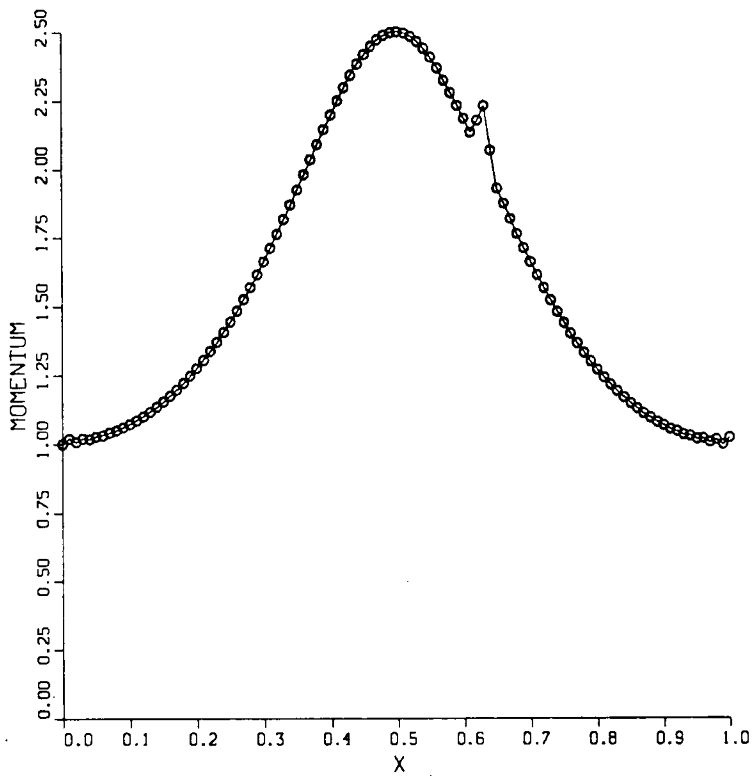

Figure 17. Steady state momentum variation.

The effect of artificial dissipation was investigated as follows. Figures 18-19 show the Froude number and x-momentum distribution for the reduced levels β={0.1, 0.1, 0.0}, and Chezy coefficient held at 1.E+08. Figures 20 - 21 show the *Fr* and x momentum distribution for β ={0.2, 0.2, 0.0}. Because the solution for ß = 0.2 showed only slight oscillations, the remainder of the problems were run utilizing a beta set of (0.25, 0.25, 0.25).

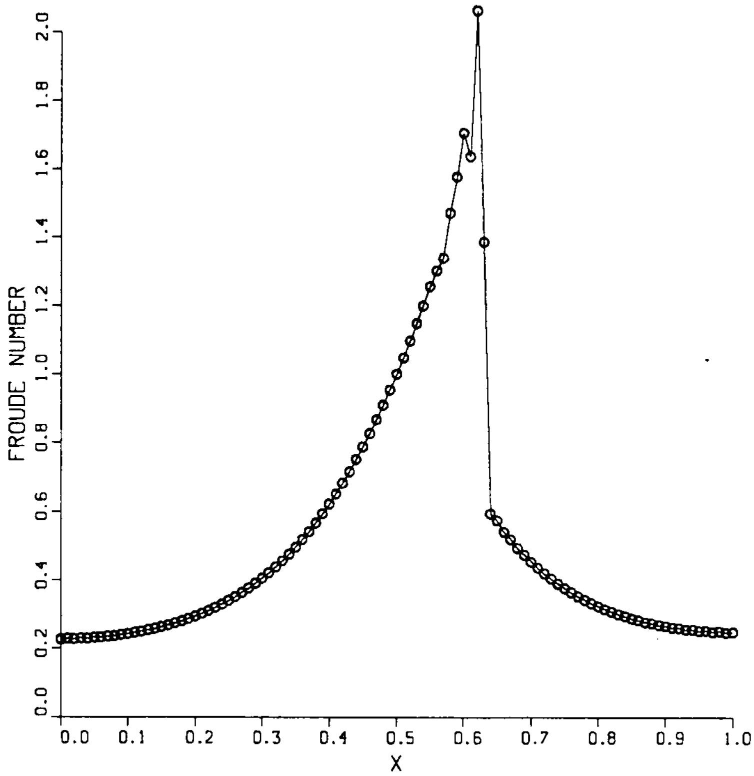

Figure 18. Steady state Froude number variation, β={0.1, 0.1, 0.0}.

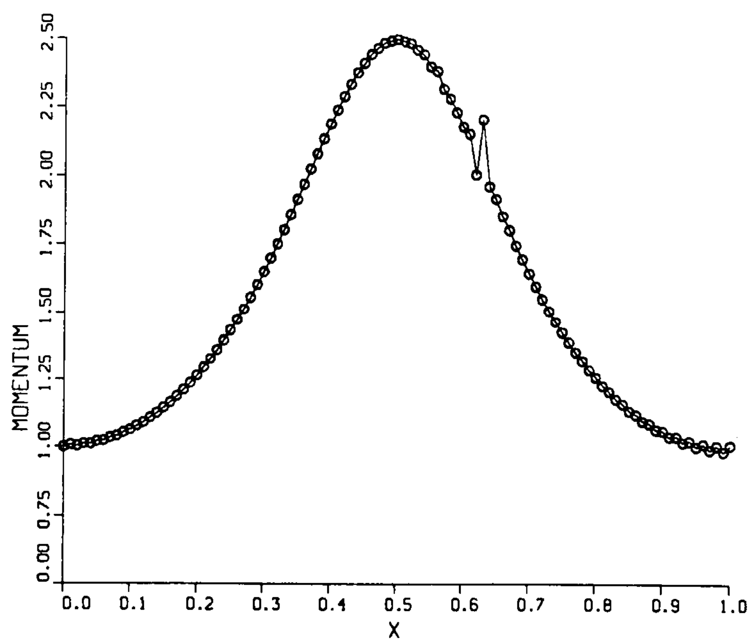

Figure 19. Steady state momentum variation, β={0.1, 0.1, 0.0}.

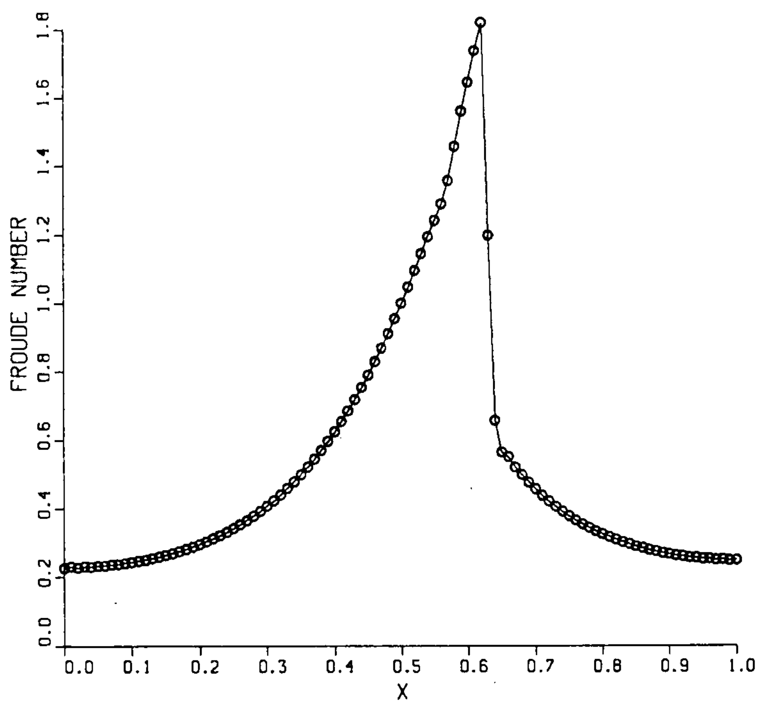

Figure 20. Steady state Froude number variation, β={02, 0.2, 0.0}.

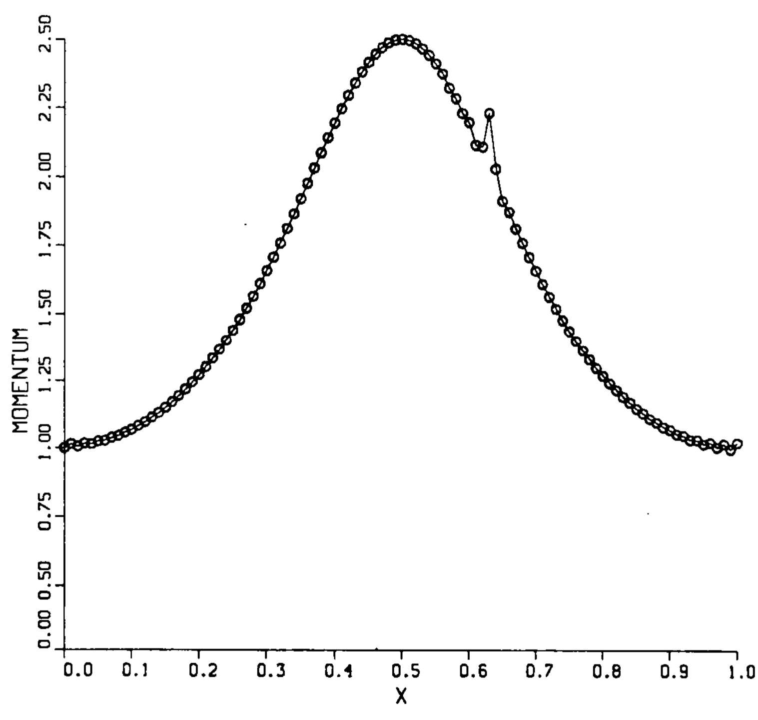

Figure 21. Steady state momentum variation, β={02, 0.2, 0.0}.

To determine the effects of bed drag, the Chezy coefficient was lowered to a value of 75. The Froude number and x-momentum distribution are plotted in Figures 22 - 23. The jump has retreated toward the throat, as less energy is available to "push back" the deeper sub-critical flow. The jump now is spread over three or four elements instead of two, as in the zero bed drag problem. Also notice the momentum overshoot at the discontinuity is greatly abated. The bed drag was further reduced to a value of C_g= 50 (Figures 24 - 25). This resulted in a fully sub-critical flow demonstrating very slight inlet and outlet oscillations. One reason for this is that the initially imposed super-critical boundary conditions were too stringent for the resulting sub-critical flow. Upon lowering the Chezy coefficient to 10, a solution flooded with the dispersion error-type waves resulted. The friction losses are now greater than the inlet momentum and hence the specified boundary conditions cannot be maintained.

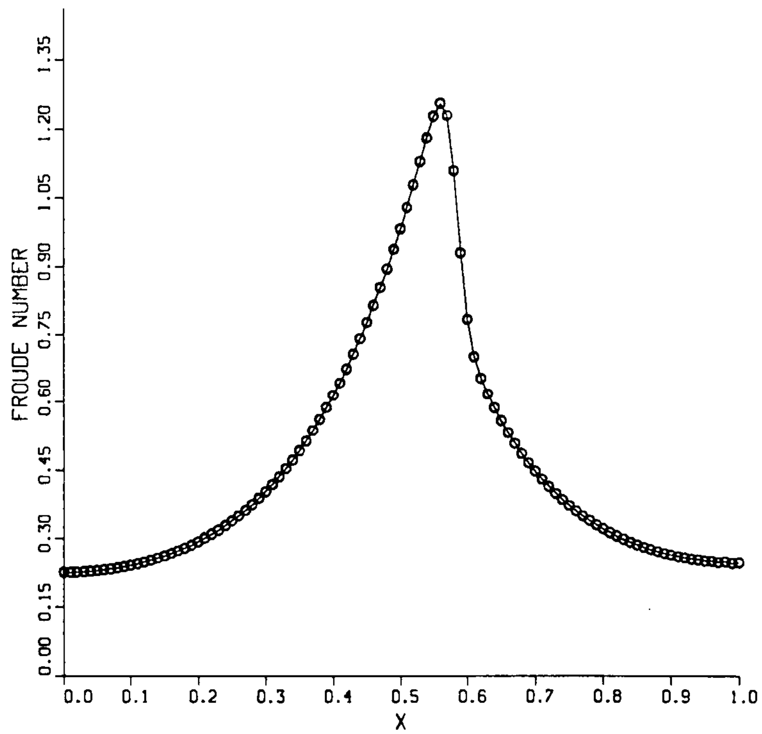

Figure 22. Steady state Froude number variation, Chezy coefficient = 75.

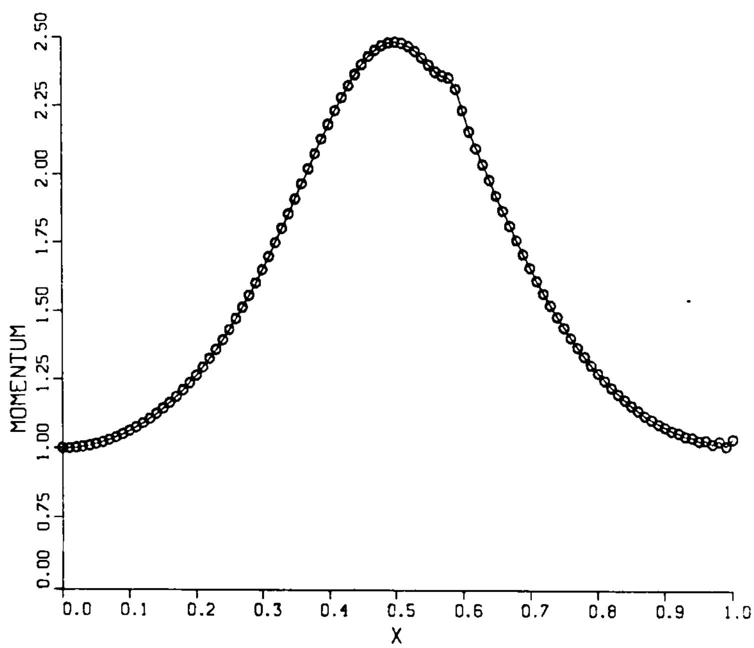

Figure 23. Steady state momentum variation, Chezy coefficient = 75.

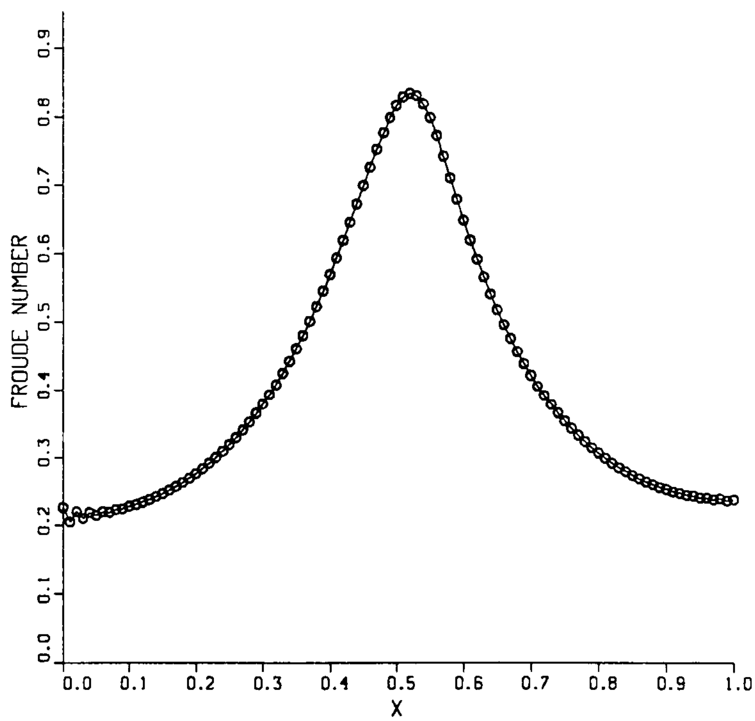

Figure 24. Steady state Froude number variation, Chezy coefficient = 50.

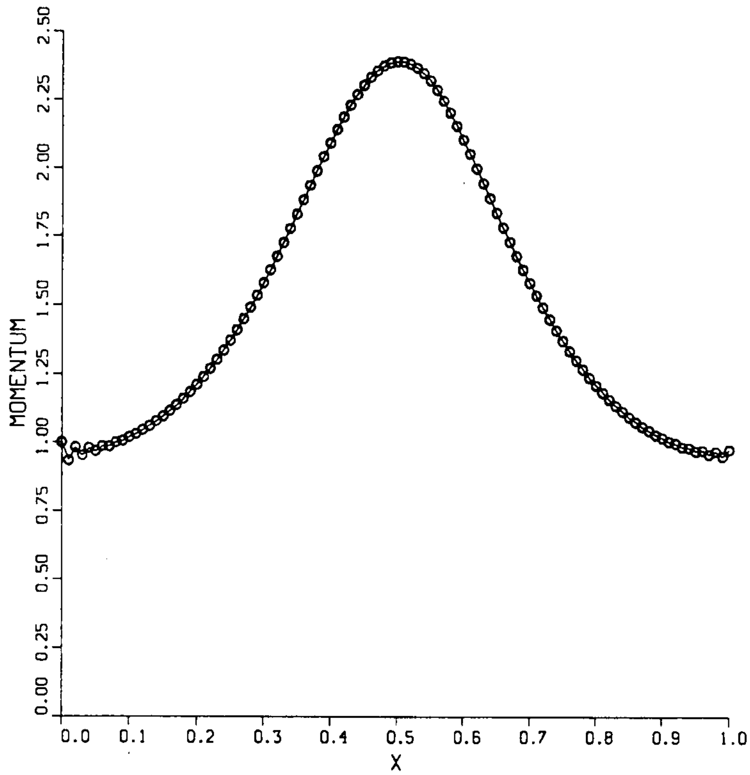

Figure 25. Steady state momentum variation, Chezy coefficient = 50.

The subsequent test case was to produce a fully super-critical exit region from the solution shown in Figures 26 -29. This requires a reduction in the free surface height to about 0.179 meters. The free surface height of the exit was first reduced to 0.5, fearing that too great a disruption from the steady-state condition might lead to a divergent solution. As expected, the location of the jump moves downstream. The further reduction of the outlet height to 0.179 allowed the jump to exit the flow domain smoothly. The outlet Froude number of 3.06 is within 3.0% of the analytical value of 3.13. However, reducing the outlet pressure to the critical state in one step was also successful. The effect of reducing the artificial dissipation is to increase the outlet Froude to a value even closer to 3.13. However, this allowed numerical oscillations to appear at the inlet and outlet boundaries.

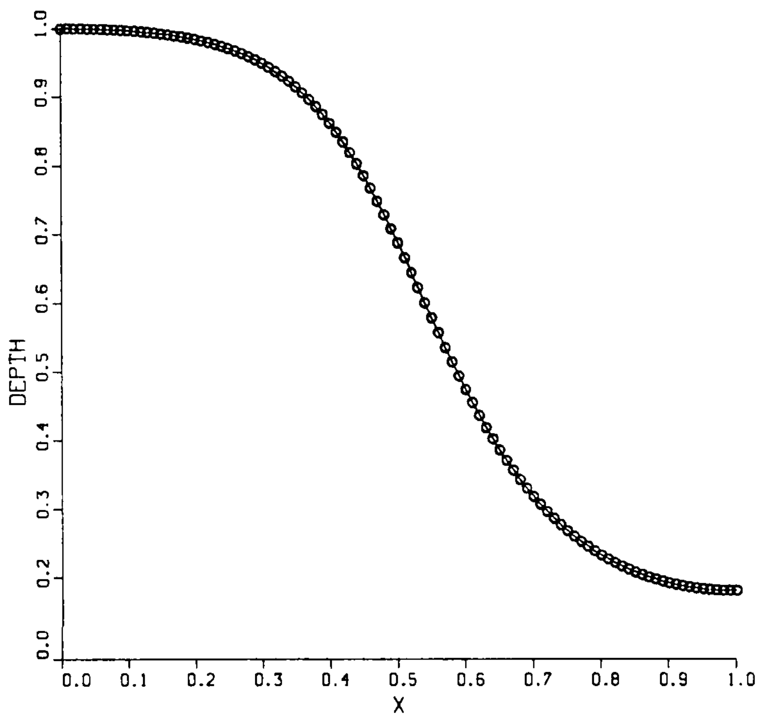

Figure 26. Steady state depth variation for the super-critical outlet.

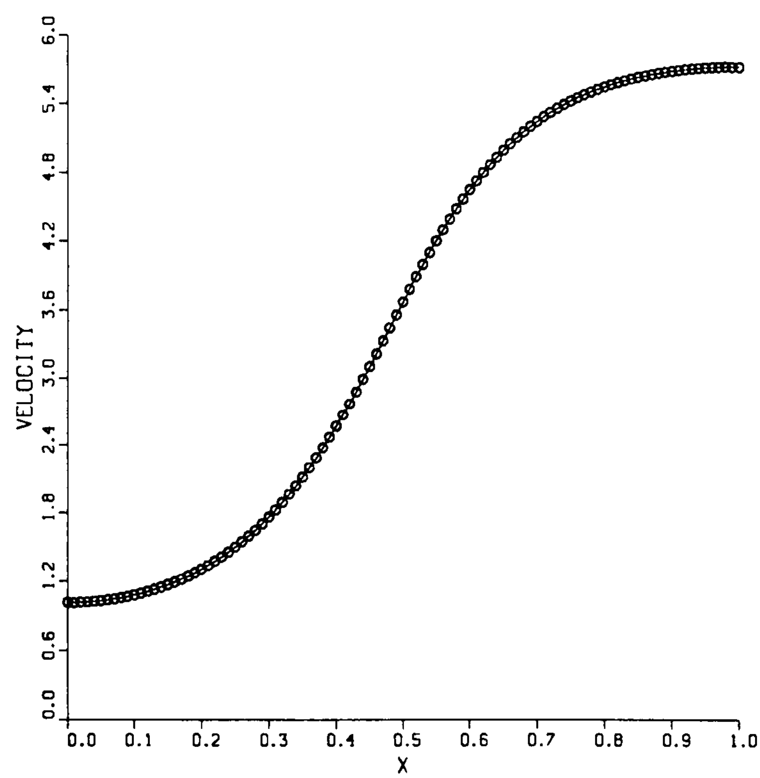

Figure 27. Steady state velocity variation for the super-critical outlet.

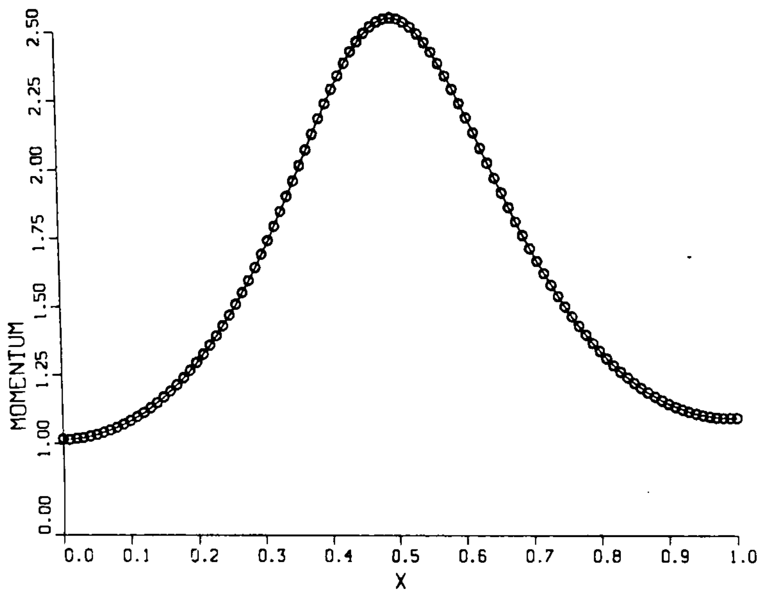

Figure 28. Steady state momentum variation for the super-critical outlet.

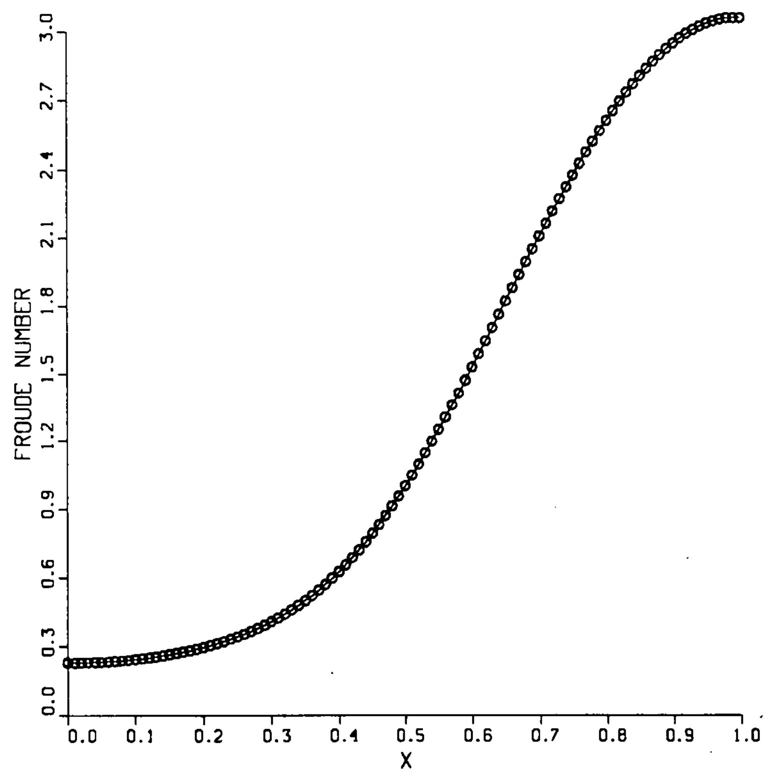

Figure 29. Steady state Froude number variation for the super-critical outlet.

The Single Rectangular Spur Dike Problem

The third problem is a straight, uniform-depth channel having a uniform width except for a single slender spur dike protruding one half the channel width and perpendicular to the walls, Figure 30.

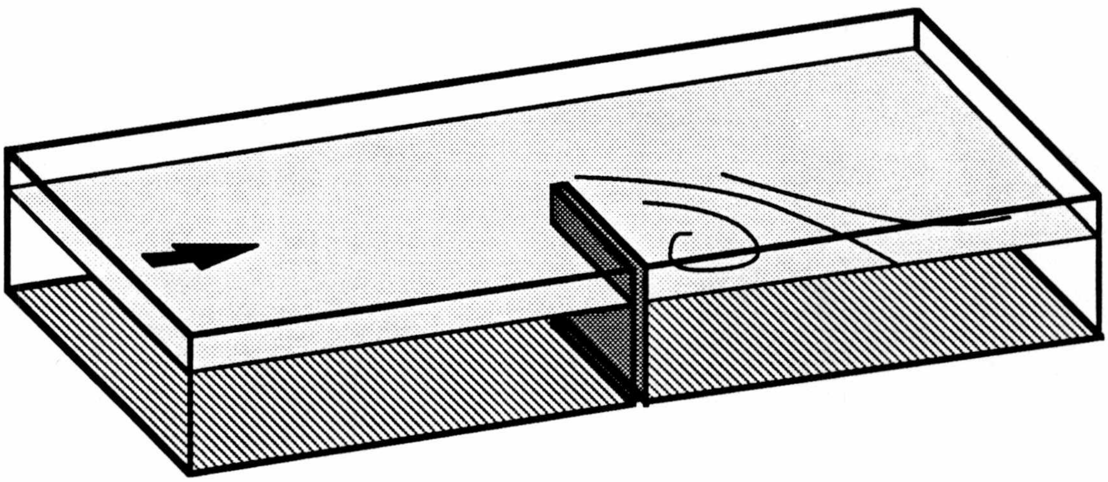

Figure 30. Geometry for the single spur dike training structure problem.

Boundary condition specification includes: fixed product m_1=hu and m_2=hv across the inlet; zero velocity perpendicular to walls (inviscid slip wall), outlet pressure fixed and vanishing normal gradients of m_1 and m_2. The modestly non-uniform mesh is twenty elements wide and sixty-three elements long, Figure 31. The elements are uniform in width and concentrated around the spur dike axially. The inlet Froude number is 0.2, initial inlet height is twenty feet and dissipation coefficients β_1, β_2 and β_3 are (0.3, 0.3, 0.3).

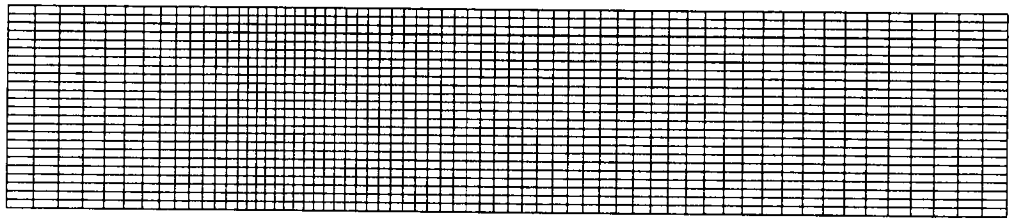

Figure 31. The rectangular spur dike problem grid.

The solution predicted by the FATHOM code, as does the analytical solution, has no steady state. Vortices are produced at the tip of the spur dike and shed downstream as a subsequent vortex builds behind it. Presented in Figure 32 is a free surface height plot at a typical instant in time. Figure 33 is a color velocity vector plot at the same instant in time. The non-dimensional celerity scale runs from zero through 1.0 . Three vortices are evident, and each is of different strength and spacing.

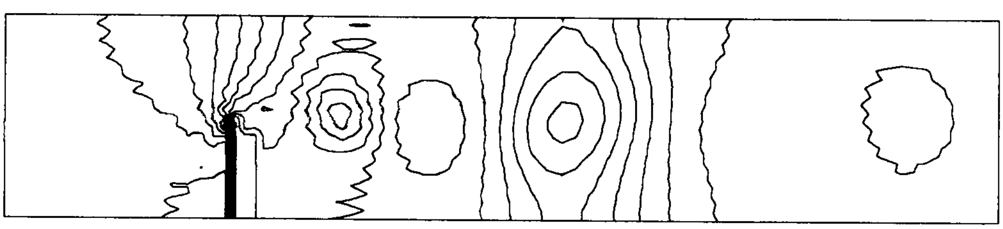

Figure 32. Height solution for the rectangular spur dike problem.

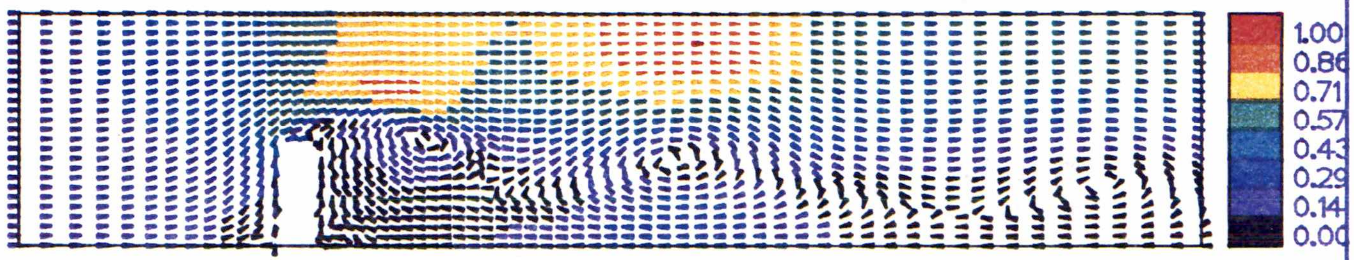

Figure 33. Velocity vector solution for single rectangular spur dike.

The dispersion-type errors in the free surface height, although small relative to the dominant flow features, were persistent during the entire analysis. A study was undertaken to ascertain the most appropriate values for the beta set. The β_1 parameter was incremented from 0.0 to 2.0 in steps of 0.1. All others were set to zero. Significant oscillations occurred for $\beta_1 < 0.2$. All values from about 0.3 - 2.0 showed about equal effectiveness. Velocity dissipation parameters, β_2 and β_3 were also examined setting the others to nought. Solutions were nominally improved for $0.1 < \beta < 0.3$. For $\beta_2 > 0.5$, oscillations led to divergence. The values chosen in Figures 32 - 33 are $\beta = (0.3, 0.3, 0.3)$. See Appendix I for details.

The velocity solution was quite monotone except for very small regions in the immediate vicinity of the corners of the spur dike, which are singular for inviscid flow. The effectiveness of the pressure outflow boundary condition is evident in that the vortex passes smoothly out (i.e., reverse flow into the outlet) of the domain.

The Triangular Spur Dike Problem

This is very similar to the previous example except for the single dike being of triangular cross section, Figure 34.

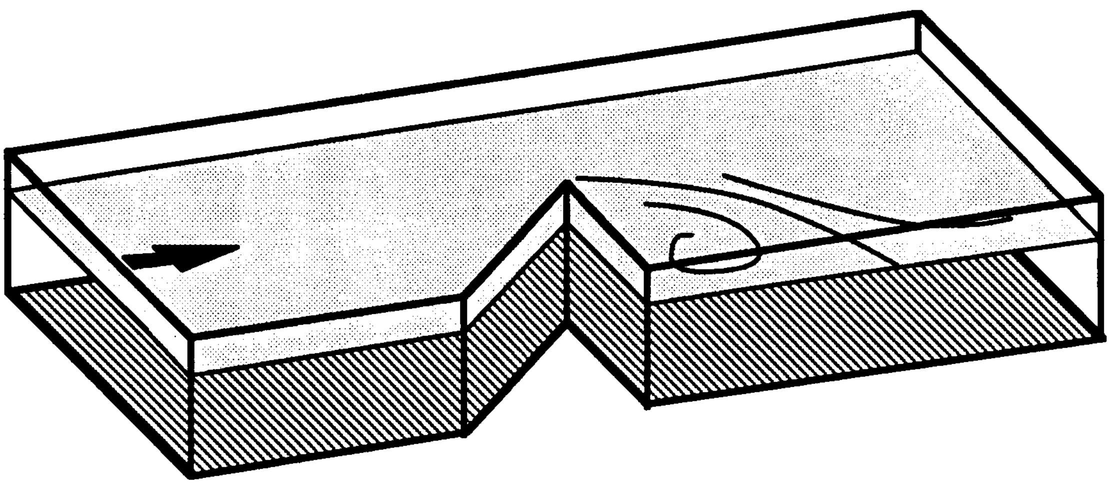

Figure 34. Geometry for the triangular training structure problem.

The modestly non-uniform mesh contains twenty elements in width and sixty-three elements axially, with local refinement around the dike axially. The only difference from the previous grid is the obvious line of non-uniformity running from the lower boundary of the mesh. The training structure generator automatically alters the grid as discussed in Chapter 3, to conform to the inclined wall of the dike, Figure 35. In addition to a more accurate enforcement of the tangential boundary condition, this allows doubling the number of grid points along this critical boundary segment. The inlet Froude number is 0.2 and inlet flow depth is twenty feet.

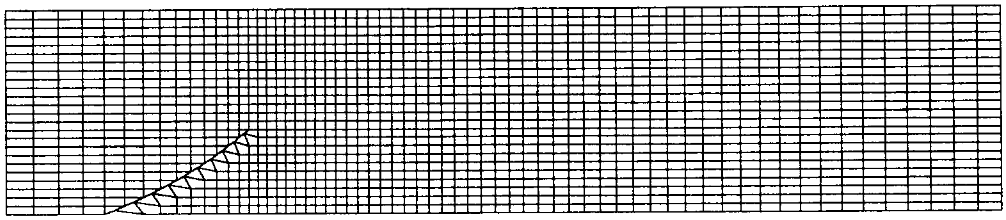

Figure 35. The triangular spur dike problem grid.

The solution predicted by the finite element procedure, again, has no steady state. Vortices produced at the tip of the dike are continually shed downstream. Figure 36 contains a snapshot of an instant illustrating a vortex located approximately midway between the dike and the outlet while another vortex nears the exit plane. The effectiveness of the pressure outflow boundary condition is again evident in that the vortex passes smoothly out of the domain.

The vortex street generated by the triangular dike is noticeably weaker and of lower frequency than that produced by the rectangular spur dike. The lower cross-stream inertia of the fluid circumventing the triangular dike creates a smaller "dead-water" region than that of the previous problem. The dispersion error "wiggles" are again present in the immediate vicinity of the spur dike tips. However, the velocity is also effected noticeably in this region. The solution shown is the best obtained after modifying the beta coefficients over the appropriate range.

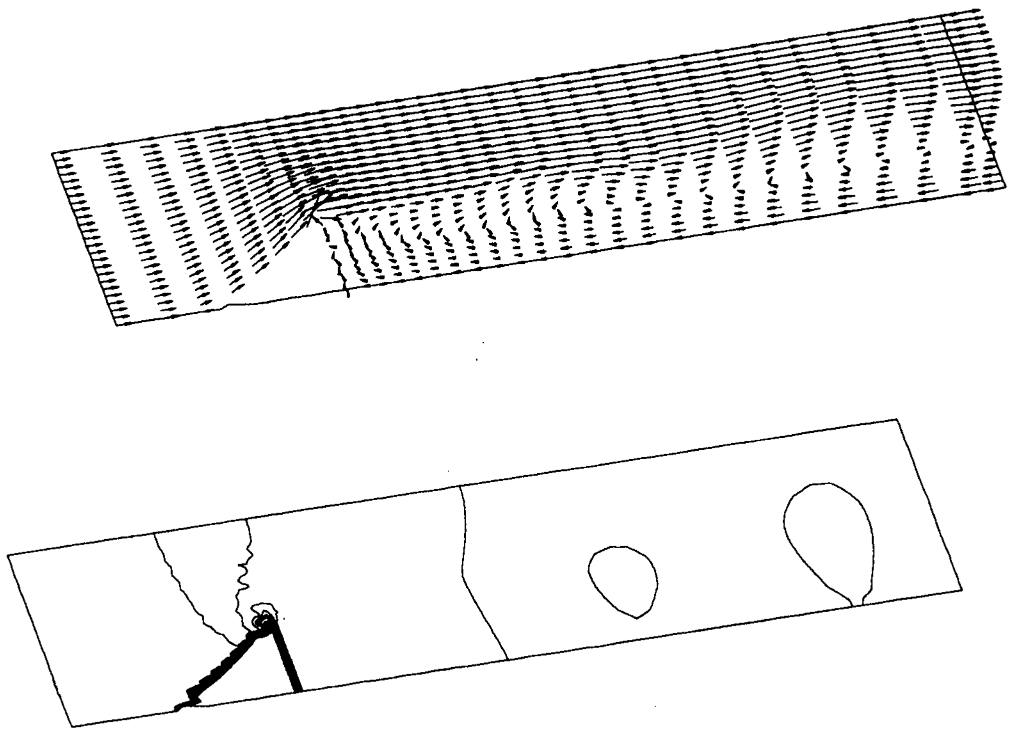

Figure 36. The triangular spur dike problem solution.

The River Bend Problem

This problem is selected as a fully viscous (zero wall velocity, finite Reynolds number) simulation of flow along a river meander. The uniform mesh shown in Figure 37 contains 23 elements in width and 95 elements axially. The lateral displacement of the meander centerline is one fourth its length. The curved sides are created as cubic spline fits. The initial conditions were constant depth, parabolic u velocity profile and zero v velocity along the entire length of the channel. The boundary conditions are zero velocity along walls (no slip) and fixed product $m_1 = hu$ and $m_2 = hv$ across the inlet. The outlet was held at a specified hydrostatic pressure and β coefficients were chosen equal to 0.3. Lower values were attempted, but produced results with severe oscillations in both the free surface and velocity.

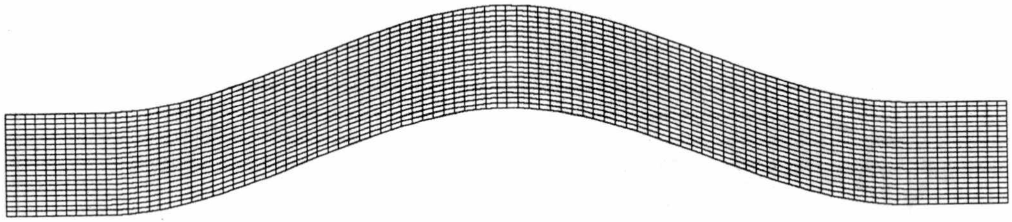

Figure 37. The river meander problem grid.

The time evolution proceeded smoothly up a point. The simulation seemed to reach a steady state, but numerical oscillations in the flow depth abruptly appeared at the outlet and invaded the entire domain. The problem was run again up to the point where the flow started to appear steady. The outlet height was then held constant at its average value and the computations continued. The solution then proceeded quickly to converge to steady state, see Figure 38.

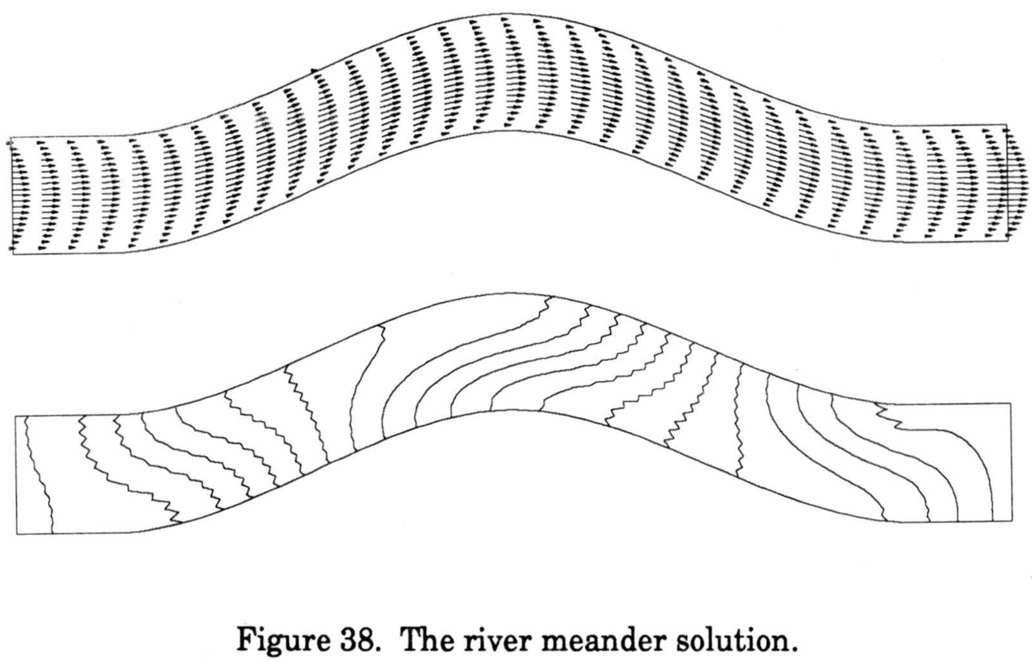

Figure 38. The river meander solution.

Slight oscillations remained in the free surface height contours, Figure 38. An attempt was made to counter these with an increase of the appropriate beta coefficient. However, a fatal corrupting of the solution occurred before the dispersion waves could be eliminated. Figure 38 represents the solution containing the lowest level of β before a notable change in the solution character occurred. The most likely cause of this numerical difficulty is the imposed boundary conditions across the inlet and outlet, known to be inconsistent with the physical solution.

The free surface height "builds up" on the outside banks and "shallows out" on the inner banks. (The inlet and outlet should allow no exception, and herein lies a known error in the boundary specification.) This behavior is known to occur in nature, promoting a secondary three dimensional flow. Note also that the maximum velocity is not on the centerline, but closer to the inner banks.

The Multiple Spur Dike Problem

This problem has a complicated geometry consisting of five spur dikes on each side of a non-uniform channel, Figure 39. The mesh contains 30 elements uniformly spaced across the flow and 120 elements axially, concentrated about each spur dike. The inlet Froude number is 0.2 and the

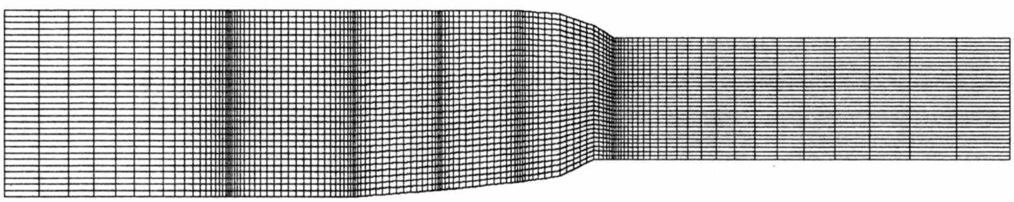

Figure 39. The multiple spur dike problem grid.

inlet depth is twenty-five feet at the shoreline. Two bed profiles were investigated. The first was horizontal (inlet depth constant at twenty-five feet). The second was similar to the southwest pass of the Mississippi River with bathymetric data modelling actual depth soundings. Boundary conditions are fixed product $m_1=hu$ and $m_2= hv$ across the inlet, hydrostatic pressure along the outlet and wall flow tangency. A uniform eddy viscosity distribution is assumed such that the effective Reynolds number is 1000. The initial condition is shown, Figure 40. Transverse velocity is zero. Axial velocity varies according to the quasi one dimensional analysis obtained from channel width, inlet height and Froude number. The beta parameter levels were found most effective at (0.3, 0.3, 0.3).

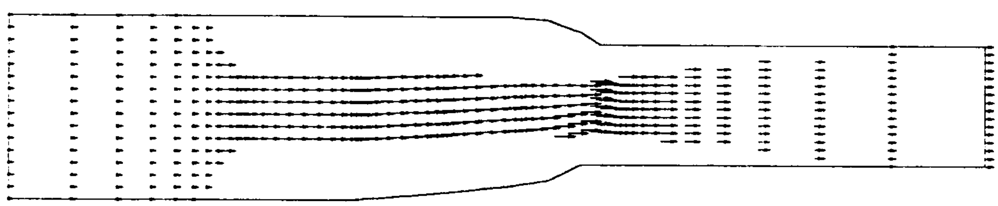

Figure 40. The multiple spur dike problem initial condition.

The simulation was run for several thousand time steps and the resulting velocity solution is shown in Figure 41. Although the velocity is quite monotone, the free surface height possesses notable oscillations in the high-velocity outlet throat. These are most likely due to the training structure corner singularities. the outflow region was unrealistically constrained in respect to the actual flow geometry constrained by the parallel wall specification o the outlet. Hence the geometry was altered to that shown in Figure 42 to better simulate flow into a large water body (the Gulf of Mexico).

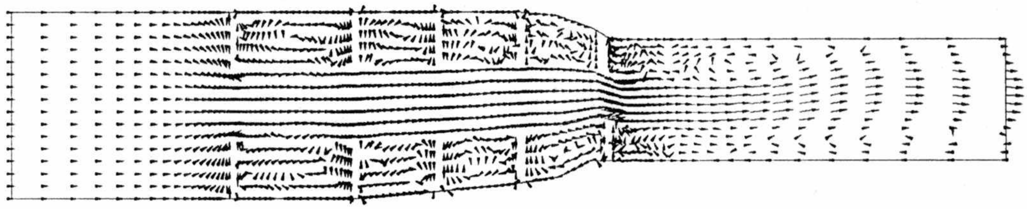

Figure 41. The multiple spur dike problem velocity solution.

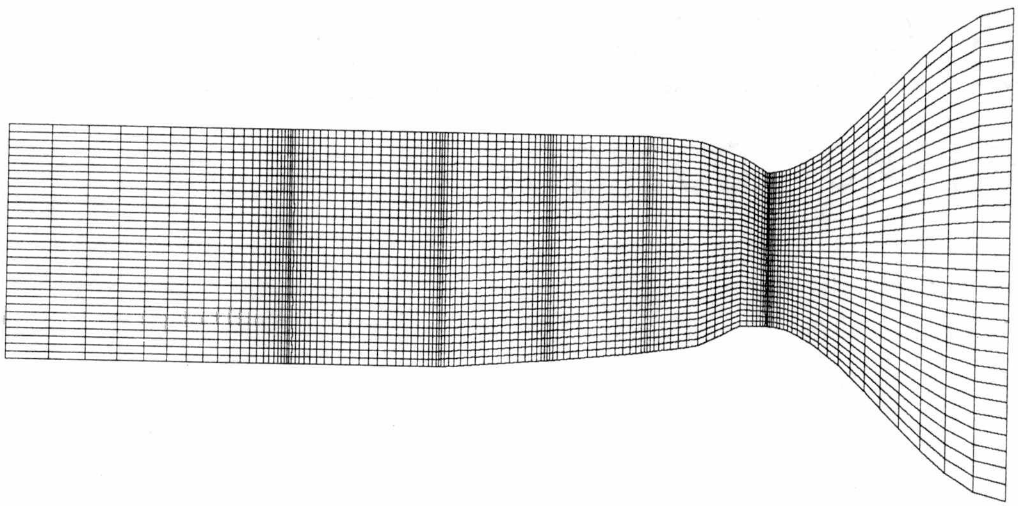

Figure 42. The multiple spur dike grid revised to incorporate a wide outlet.

Changing the boundary geometry necessitates a corresponding alteration of the flow solution along the boundaries in order to ensure wall tangency and hence convergence. To accomplish this in a straight-forward systematic manner, the velocity along the two walls of the exit region is manually set to zero and the time step reduced. Manually adjusting the wall velocity requires determining the global numbers of the wall nodes and editing the corresponding lines in the FOR061.DAT file. Internal nodes were unaltered. The new problem was run successfully to that shown in Figure 43. Some oscillations did occur in the free surface depth and velocity, however, they were not serious enough to lead to a divergent solution after several hundred time steps. No steady state solution exists, but Figure 43 is representative. This instant is chosen because of its low level of visible dispersion error in the relatively slow recirculation regions (where small disturbances more visibly disrupt the flow field).

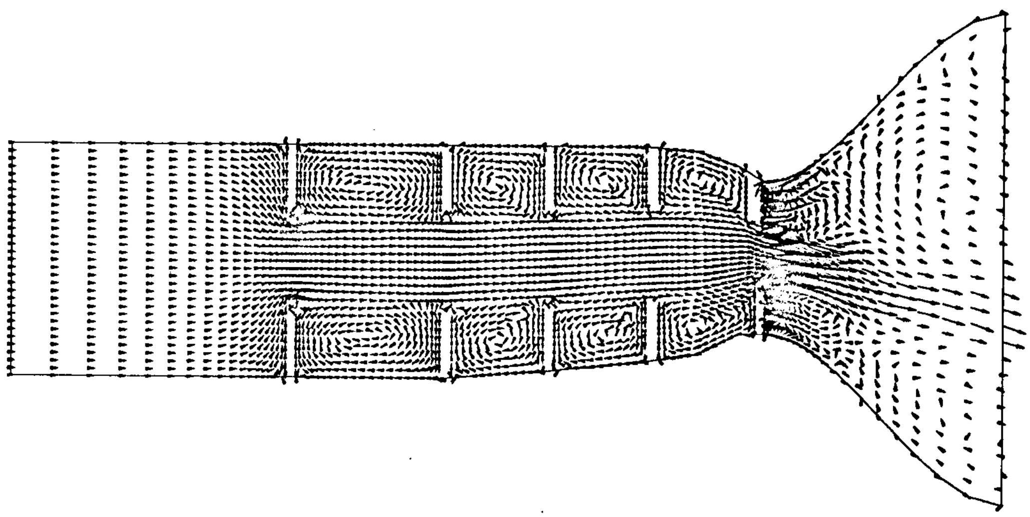

Figure 43. The wide outlet multiple spur dike velocity solution.

In order to run a similar problem involving a realistic shipping channel, the desired bed profile is input through the function BEDEL in the FEMFSI program. The grid employed is identical to the one used in Figure 39. A pictorial view of the bed profile and spur dike height is shown, Figure 44. The top of the spur dikes correspond to the initial (horizontal) free surface. Initial depth along each bank is 20 feet and at the center of the channel is 40 feet. The inlet Froude number is 0.2 and the effective Reynolds number is 1000. Beta levels were set to {0.3, 0.3, 0.3}. A representative output of the time dependent velocity and free surface height is illustrated in Figure 45 - 46.

The overall character of the solution is similar to the previous problem. The most obvious difference is the increase of dispersion-type error in both the free surface height and velocity. A source of errors is the slope discontinuity at the top and bottom of the channel. The magnitude of the dispersion-induced velocity is on the order of 3% of the axial velocity. Of great practical importance is the increase in velocity attained by adding the bed channel. Centerline velocity increases by 43% for this simulation. This is of value since dikes are built in order to decrease sedimentation by increasing kinetic energy through the shipping channel.

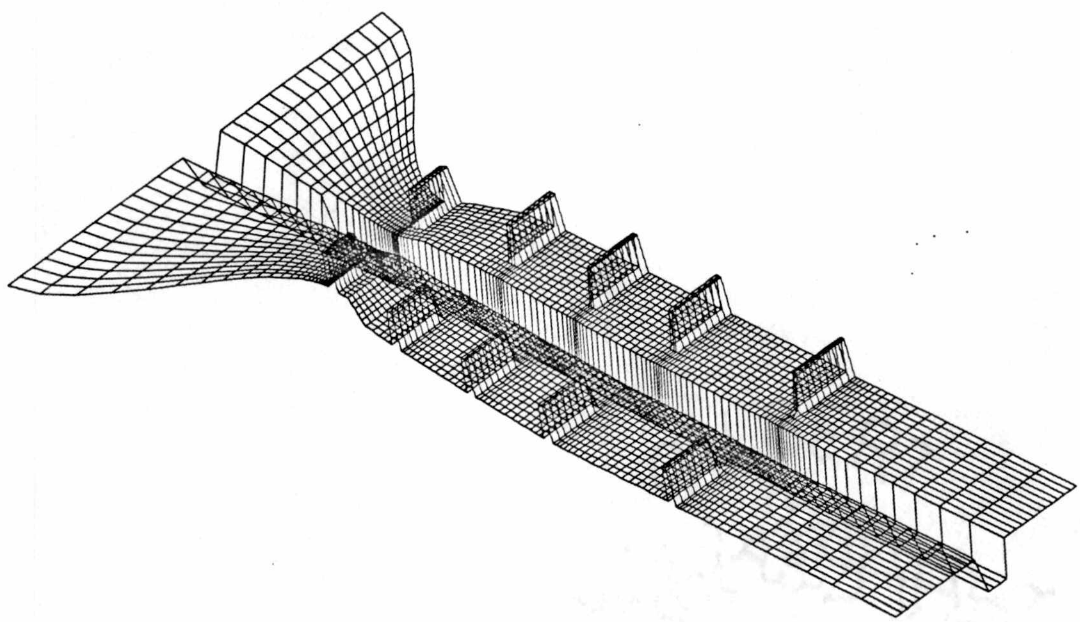

Figure 44. The wide outlet bed profile and spur dike description.

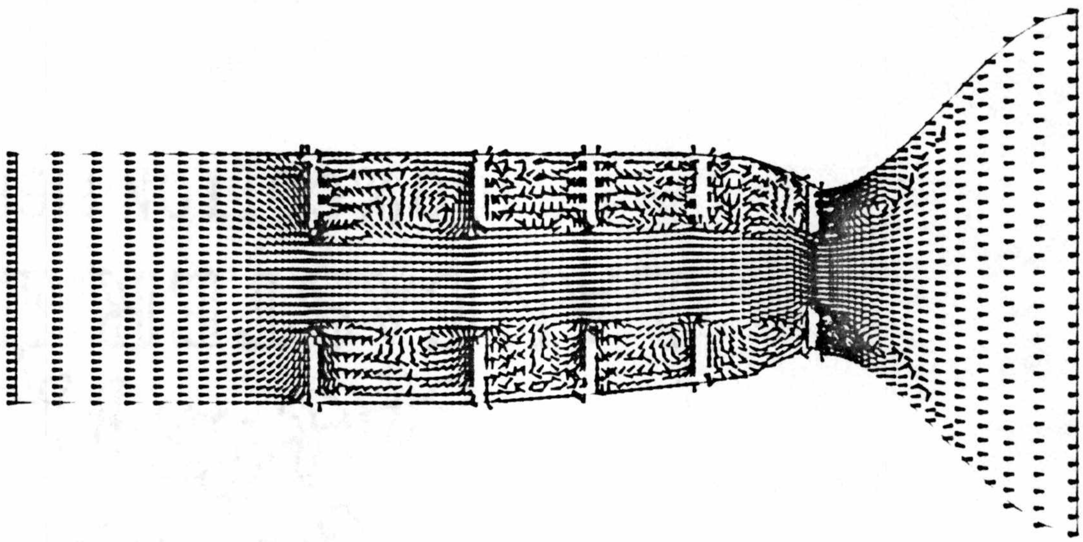

Figure 45. The Mississippi River outlet simulation velocity solution.

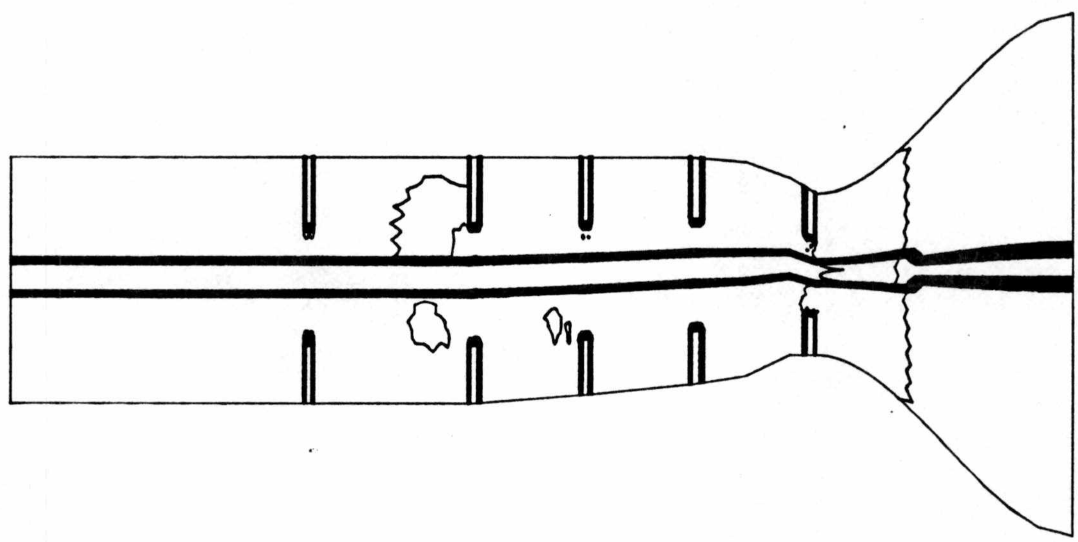

Figure 46. The Mississippi River outlet simulation depth solution.

LIST OF REFERENCES

[1] Baker, A. J. *Finite Element Computational Fluid Mechanics*, Hemisphere, Washington, D.C. (1983).

[2] Baker, A. J., Iannelli, G. S. and Manhardt, P. D. "A Taylor Weak Statement Finite Element CFD Algorithm for TABS-3," COMCO Technical Report 88-25.1, (1988)

[3] Baker, A. J. and Kim, J. W., "A Taylor Weak Statement Algorithm for Hyperbolic Conservation Laws," Int. J. Num. Mtds/ Fluids, 7(1987).

[4] Brebbia, C. A. and Partridge, P. W. , "Finite Element Simulation of Water Circulation in the North Sea," J. Appl. Math Modelling, Vol. 1, (1976).

[5] Chakravarthy, S., "Euler Equations - Implicit Schemes and Implicit Boundary Conditions," AIAA Paper 82-0223 (1982).

[6] Iannelli, G. S., Private communication, Knoxville, (1988).

[7] Iannelli, G. S. and Baker, A. J., An Efficient Solution Adaptive Implicit Finite Element CFD Navier Stokers Algorithm," AIAA Paper 90-0400 (1990).

[8] Iannelli, G. S. and Baker, A. J., An Implicit and Stiffly Stable Finite Element CFD Algorithm for Unsteady Aerodynamics," AIAA Paper 89-0656 (1989).

[9] King, I. P., "A Model for Three Dimensional Density Stratified Flow," Resource Management Associates, Lafayette, CA (1988).

[10] White, F. M., Viscous Fluid Flows, McGraw-Hill, New York (1974).

APPENDICES

APPENDIX A

Specification of Longitudinal Troughs

The user can supply the key data describing longitudinal depth distributions. These are obtained by segmenting the channel bed into suitable "longitudinal regions." Each region is composed of four edges in order to accommodate longitudinal troughs, see Figure A.1.

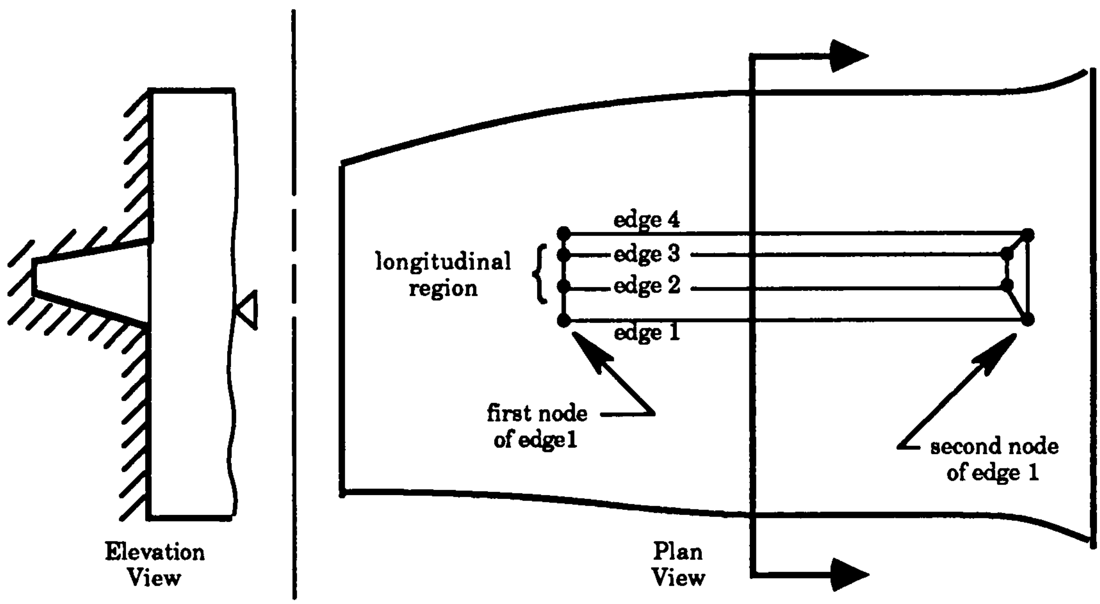

Figure A-1. Example of a longitudinal trough in the flow field.

After supplying the channel's upper and lower bank depths, the user is requested to input, for every region, the coordinates of the first and second node of each edge along with the depth along that line.

APPENDIX B

The Implicit Taylor's Series

The Taylor's series about time t, in the time interval $t_{n+1} \geq t \geq t_n$, of the function q, to first order accuracy is

$$q = q_n + (t - t_n)\frac{\partial q}{\partial t}\bigg|_n + O(\Delta t^2)$$

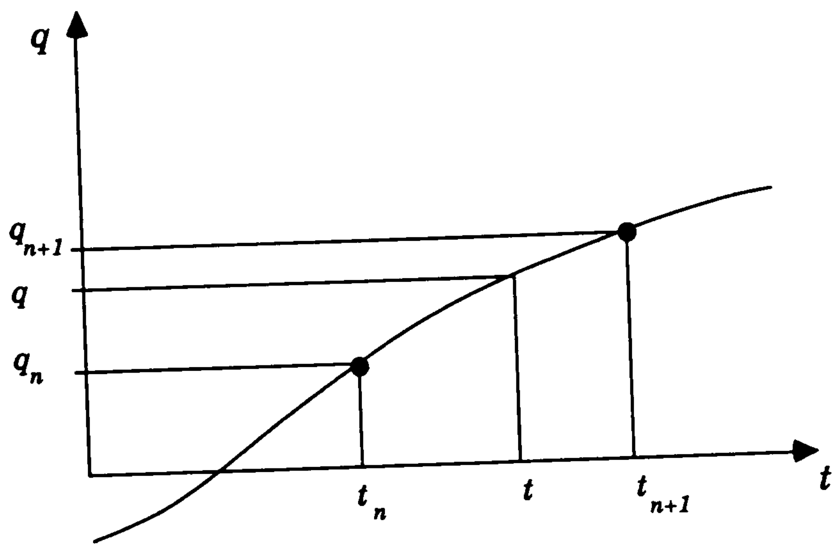

Figure A-2. Value of the function q during a typical time interval.

Alternately, q may be determined from the time t_{n+1} to be

$$q = q_{n+1} - (t_{n+1} - t)\frac{\partial q}{\partial t}\bigg|_{n+1} + O(\Delta t^2)$$

Hence, equating the two yields (deleting the terms of second order and above)

$$q_n + (t - t_n)\left.\frac{\partial q}{\partial t}\right|_n = q_{n+1} - (t_{n+1} - t)\left.\frac{\partial q}{\partial t}\right|_{n+1}$$

or

$$q_{n+1} = q_n + (t - t_n)\left.\frac{\partial q}{\partial t}\right|_n + (t_{n+1} - t)\left.\frac{\partial q}{\partial t}\right|_{n+1}$$

Dividing the entire expression by the total time interval $(t_{n+1} - t_n) = \Delta t$

$$\frac{q_{n+1}}{\Delta t} = \frac{q_n}{\Delta t} + \frac{(t - t_n)}{(t_{n+1} - t_n)}\left.\frac{\partial q}{\partial t}\right|_n + \frac{(t_{n+1} - t)}{(t_{n+1} - t_n)}\left.\frac{\partial q}{\partial t}\right|_{n+1}$$

Defining the implicitness parameter, θ,

$$\theta \equiv \frac{(t_{n+1} - t)}{(t_{n+1} - t_n)}$$

and multiplying by Δt throughout yields the desired form

$$q_{n+1} = q_n + \Delta t\left((1 - \theta)\left.\frac{\partial q}{\partial t}\right|_n + \theta\left.\frac{\partial q}{\partial t}\right|_{n+1}\right)$$

APPENDIX C

Enforcement of the Wall Tangency Boundary Condition

The enforcement of the wall tangency boundary condition is effected by deleting the appropriate surface integral terms in the weak statement, hence

$$\boldsymbol{m} \bullet \boldsymbol{n} = 0$$

where $\boldsymbol{n}$ is the outward pointing normal unit vector to the boundary segment and $m \equiv uh$ is the depth flux. However, this formulation has been proven insufficient, although necessary. Conservation of mass is found not to be enforced at corner singularities along spur dikes. To counter this, the time derivative of the momentum flux is set to zero along such boundaries. This yields a significant improvement over previous methods. The formulation is transparent to the operation of the code.

APPENDIX D

Changing Program Dimensions

FATHOM has been written with ease of use and maintainability in mind. Hence, the dimensions of the various arrays are easily altered by changing the appropriate dimension statements at the beginning of the program(s). There are, however, some variables which should only be changed in groups, since they necessarily utilize a single parameter. The following list covers all groups which may be changed to increase or decrease the amount of allocated storage. For example, SMAT(I,*), F(I) implies that the first dimension of SMAT and the dimension of F must be identical.

For the GENGRID and FILEGRID:

SMAT(I,*), F(I)

QA(J), Q(J), XT(J), YT(J)

SRX(K), SRY(K), WD(K), LEP(K)

COORD(K,2), COORX(K,2), IBA(K,10), NODLP(K,4)

NPBCV(K), NPBCX(K)

LNODS(L,9), LNODX(L,9)

NBNO(M),NCOD(4*M), NBNX(M)

IBN(M),A(M),B(M)

IBL(N), IBAL(N), IBA(N), NODLL(N,2)

For FEMFSI and FEMFILE:

same as for FEMFRS, but add

NODEC(M), IBN(M)

For TSGEN:

LNODS(L,4), X(L), Y(L)

NCPBV(L), Q(4*L)

NBNB(M),NBNT(M)

ISP(P),INCO(P,2,2)

For FEMFRS:

SMAT(I,*), F(I)

QA(J), Q(J), XT(J), YT(J)

SRX(K), SRY(K), WD(K), LEP(K)

COORD(K,2), COORX(K,2), IBA(K,10), NODLP(K,4)

NPBCV(K), NPBCX(K)

LNODS(L,9), LNODX(L,9)

NBNO(M),NCOD(4*M), NBNX(M)

IBL(N), IBAL(N), IBA(N), NODLL(N,2)

APPENDIX E

A Tutorial to GENGRID

```
Do you wish to generate a new grid or view a previous grid?
   0  >>  view old grid only
   1  >>  create a new grid
   1

INPUT TOTAL NUMBER OF COLUMNS ?
19
TOTAL NUMBER OF ROWS ?
7
YOU HAVE           133 ELEMENTS
WHICH FORM       160 NODES ALTOGETHER
                        52   OF WHICH ARE BOUNDARY NODES.

Y CO-ORDINATE ZONES TOTAL NUMBER OF ZONES?
2
ANY Y-BOUNDARY CUSPS ? (1=YES)
0
X-CO-ORDINATE ZONES
TOTAL NUMBER OF ZONES ?
2
ANY X-BOUNDARY CUSPS ?  (1=YES)
1
HOW MANY CUSPS ?
1

    * BOUNDARY GEOMETRY *
HOW MANY BOUNDARY SECTIONS ?
5
X CO-ORDINATE CUSP COLUMNS
CUSP1 COLUMN INDEX ?
10

WHICH ROW ?
1< FIRST ROW
2< LAST  ROW
1

X-CO-ORDINATE-ZONE COLUMNS
ZONE             1
LAST-ROW INDEX ?
10

ZONE BOUNDARY-SHAPE SPECIFICATIONS?
SIDE             1
ORTHOGONAL          CODE: 1
ANGLE PRESCRIBED    CODE: 2
1
```

```
SIDE              2
ORTHOGONAL          CODE: 1
ANGLE PRESCRIBED   CODE: 2
1

FINAL NODE OF SECTION:            1
LOCATION BY ELEMENT CO-ORDINATES:
ELEMENT ROW ?
1
ELEMENT COLUMN ?
10
LOCAL NODE NUMBER ?
2

ELEMENT              64                    NODE              81

FINAL NODE OF SECTION:            2
LOCATION BY ELEMENT CO-ORDINATES:
ELEMENT ROW ?
1
ELEMENT COLUMN ?
19
LOCAL NODE NUMBER ?
2

ELEMENT             127                    NODE             153

FINAL NODE OF SECTION:            3
LOCATION BY ELEMENT CO-ORDINATES:
ELEMENT ROW ?
7
ELEMENT COLUMN ?
19
LOCAL NODE NUMBER ?
3

ELEMENT             133                    NODE             160

FINAL NODE OF SECTION:            4
LOCATION BY ELEMENT CO-ORDINATES:
ELEMENT ROW ?
7 ELEMENT COLUMN ?
1
LOCAL NODE NUMBER ?
4
ELEMENT               7                    NODE               8

EXTREME-NODE CO-ORDINATES
FIRST NODE CO-ORDINATES  X= ?
0.0000000E+00
Y= ?
0.0000000E+00

SECTION            1
LAST NODE CO-ORDINATES
```

```
X= ?
0.3333333
Y= ?
0.2000000

SECTION             2
LAST NODE CO-ORDINATES
X= ?
1.000000
Y= ?
0.0000000E+00

SECTION             3
LAST NODE CO-ORDINATES
X= ?
1.000000
Y= ?
0.5000000

SECTION             4
LAST NODE CO-ORDINATES
X= ?
0.0000000E+00
Y= ?
0.5000000

SECTION SHAPES
SECTION             1
X<CONST  Y<VAR              CODE: 1
X<VAR    Y<CONST            CODE: 2
X<VAR    Y<VAR , Y<IND      CODE: 3
X<VAR    Y<VAR , X<IND      CODE: 4
4

 I) LINEAR      Y   CODE: 1
II) QUADRATIC   Y   CODE: 2
III) CUBIC       Y   CODE: 3
IV) CUBIC SPLINE    CODE: 4
1

SECTION             2
X<CONST  Y<VAR              CODE: 1
X<VAR    Y<CONST            CODE: 2
X<VAR    Y<VAR , Y<IND      CODE: 3
X<VAR    Y<VAR , X<IND      CODE: 4
4

 I) LINEAR      Y   CODE: 1
II) QUADRATIC   Y   CODE: 2
III) CUBIC       Y   CODE: 3
IV) CUBIC SPLINE    CODE: 4
1
```

```
SECTION              3
X<CONST   Y<VAR              CODE: 1
X<VAR     Y<CONST            CODE: 2
X<VAR     Y<VAR , Y<IND      CODE: 3
X<VAR     Y<VAR , X<IND      CODE: 4
1

SECTION              4
I) LINEAR       Y    CODE: 1
II) QUADRATIC   Y    CODE: 2
III) CUBIC      Y    CODE: 3
IV) CUBIC SPLINE     CODE: 4
2

SECTION              5
I)    LINEAR      Y    CODE: 1
II) QUADRATIC     Y    CODE: 2
III) CUBIC        Y    CODE: 3
IV) CUBIC SPLINE       CODE: 4
1

POINT-DISTRIBUTION
SECTION              1
GRID POINT UNIFORMITY:
UNIFORM       DISTRIBUTION     CODE: 1
LEFT   SIDE CONCENTRATION      CODE: 2
RIGHT SIDE CONCENTRATION       CODE: 3
BOTH   SIDE CONCENTRATION      CODE: 4
CENTER        CONCENTRATION    CODE: 5
3
STRETCHING STRENGTH   (1 THROUGH 20)
1.200000

SECTION              2
GRID POINT UNIFORMITY:
UNIFORM       DISTRIBUTION     CODE: 1
LEFT   SIDE CONCENTRATION      CODE: 2
RIGHT SIDE CONCENTRATION       CODE: 3
BOTH   SIDE CONCENTRATION      CODE: 4
CENTER        CONCENTRATION    CODE: 5
2
STRETCHING STRENGTH   (1 THROUGH 20)
2.000000

SECTION              3
GRID POINT UNIFORMITY:
UNIFORM       DISTRIBUTION     CODE: 1
LEFT   SIDE CONCENTRATION      CODE: 2
RIGHT SIDE CONCENTRATION       CODE: 3
BOTH   SIDE CONCENTRATION      CODE: 4
CENTER        CONCENTRATION    CODE: 5
3
STRETCHING STRENGTH   (1 THROUGH 20)
2.000000
```

```
SECTION           4
GRID POINT UNIFORMITY:
UNIFORM      DISTRIBUTION    CODE: 1
LEFT   SIDE CONCENTRATION    CODE: 2
RIGHT SIDE CONCENTRATION     CODE: 3
BOTH   SIDE CONCENTRATION    CODE: 4
CENTER      CONCENTRATION    CODE: 5
STRETCHING STRENGTH   (1 THROUGH 20)
10.00000

SECTION           5
GRID POINT UNIFORMITY:
UNIFORM      DISTRIBUTION    CODE: 1
LEFT   SIDE CONCENTRATION    CODE: 2
RIGHT SIDE CONCENTRATION     CODE: 3
BOTH   SIDE CONCENTRATION    CODE: 4
CENTER      CONCENTRATION    CODE: 5
2
STRETCHING STRENGTH   (1 THROUGH 20)
2.000000

STORE ON FILE
YMIN=  0.0000000E+00
YMAX=  0.5000001
SELECT A DEVICE
1<< MONITOR
2<< PLOTTER 1
1

SELECT A MONITOR
1<< VT220
2<< TEK 4114
3<< TEK 4207
1
```

APPENDIX F

A Tutorial to FEMFSI

```
GRAVITATIONAL ACCELERATION ?
  9.810000

PLEASE SELECT ONE OF THE FOLLOWING INITIAL-CONDITION PARAMETER CHOICES
1<< INLET Fr #  &  AVERAGE  DEPTH     SPECIFIED
2<< INLET Fr #  &  AVERAGE  VELOCITY  SPECIFIED
3<< INLET AVERAGE VELOCITY & DEPTH    SPECIFIED
4<< INLET Fr # & STAGNATION DEPTH     SPECIFIED
5<< INLET AVERAGE & STAGNATION DEPTHS SPECIFIED
          1

 INLET FROUDE # ?
 0.1000000

 INLET AVERAGE DEPTH ?
  1.000000

 INLET AVERAGE VELOCITY =    0.6264184

 DO YOU WISH TO SPECIFY THE REYNOLDS NUMBER?
 1 << YES
 2 << NO
          1

 REYNOLDS NUMBER SPECIFICATIONS
 SELECT ONE OF THE FOLLOWING COMBINATIONS

 1 << REFERENCE VELOCITY
      REFERENCE LENGTH AND
      EDDY VISCOSITY  SPECIFIED

 2 << REFERENCE LENGTH
      REYNOLDS NUMBER AND
      EDDY VISCOSITY  SPECIFIED
          2

 REFERENCE LENGTH ?
   1.000000
 REYNOLDS NUMBER ?
   1000.000
 EDDY VISCOSITY ?
  1.0000000E-03
 REFERENCE VELOCITY=   1.000000

 GENERALIZED CHEZY COEFFICIENT ?
  1.0000000E+08
 REFERENCE FROUDE #=  0.3192754
```

```
DO YOU PREFER A QUASI 1-D ANALYSIS ?
1<< YES
2<<  NO
          1
ARE YOU STUDYING A TRAINING-STRUCTURE
PROBLEM ?
1<< YES
2<<  NO
          2
THE PROBLEM-PARAMETER  FILE HAS BEEN
CREATED

DO YOU WISH TO SET UP INITIAL CONDITIONS ?
1<< YES
2<<  NO
          1

DO YOU WISH TO SPECIFY AN AVERAGE INITIAL OUTFLOW HEIGHT ?
1<< YES
2<<  NO
          2

IS THE INITIAL DEPTH DISTRIBUTION KNOWN ?
1<< YES
2<<  NO
          2

PLEASE SELECT THE TYPE OF FLOW
1<< INVISCID FLOW WITH CONSTANT U ALONG Y
2<< VISCOUS  FLOW
3<< INVISCID FLOW WITH VARIABLE U ALONG Y
          1

DO YOU WISH TO UTILIZE THE DEPTH-SPECIFICATION ROUTINE ?
1<< YES
2<<  NO
          2

OUTFLOW HEIGHT CONSTRAINED ?
1<< YES
2<<  NO
          1

INLET FLOW DIRECTION PRESCRIBED ?
1<< YES
2<<  NO
          1

INLET FLOW DIRECTION  ? (ANGLE IN DEGREES)
 0.0000000E+00
THE INITIAL-CONDITION FILES HAVE BEEN
CREATED
```

```
DO YOU WISH TO SET UP TIME-INTEGRATION
ALGORITHM PARAMETERS ?
1<< YES
2<<  NO
         1

TYPE OF INTEGRATION ALGORITHM ?
1=EULER BACKWARD
2=CRANK-NICOLSON
3=IMPLICIT RUNGE-KUTTA 87
4=IMPLICIT RUNGE-KUTTA 88
         1

ENTER MAX. NUMBER OF NEWTON ITERATIONS (>=2)
         2
THE TIME-INTEGRATION ALGORITHM PARAMETERS
 FILE HAS BEEN CREATED

DO YOU WISH TO SET UP TIME-STEP
VARIATION PARAMETERS ?
1<< YES
2<<  NO
         1

 INITIAL TIME ?
  0.0000000E+00
 TIME MAX ?
   1000.000
 TOTAL NUMBER OF STEPS ?
          25
 MINIMUM TIME STEP ? (< INITIAL DT !)
  9.9999997E-06
 INITIAL TIME STEP ?
  1.0000000E-03
 MAXIMUM TIME STEP ?
  0.1000000
 USER DEFINED ZERO ?
  9.9999997E-06
 DT-VARIATION PARAMETER (0.0< PAR <=1.0)
  0.3000000
 THE TIME-STEP VARIATION PARAMETERS
  FILE HAS BEEN CREATED

 1ST BETA COEFFICIENT ?
  0.1000000
 2ND BETA COEFFICIENT ?
  0.1000000
 3RD BETA COEFFICIENT ?
  0.1000000
 4TH BETA COEFFICIENT ?
  0.1000000
 THE DISSIPATION MECHANISM PARAMETER FILE
 HAS BEEN CREATED
```

```
DO YOU WISH TO SET-UP CHANNEL-BED DATA ?
1<< YES
2<<  NO
          1

TYPE OF CHANNEL-BED INPUT DATA

1<< BED ELEVATION AT MESH NODES
2<< BED SLOPES    AT MESH NODES
          1
THE CHANNEL-BED DATA FILE HAS BEEN
CREATED

DO YOU WISH TO SET-UP BOUNDARY-CONDITION INFORMATION ?
1<< YES
2<<  NO
          1

 BOUNDARY CONDITIONS
HOW MANY BOUNDARY SECTIONS ? (<=20)
          8
          8 SECTIONS
NOW ENTER:  C ,TO CONFIRM
C

SECTION             1
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
          2
THE TYPE SELECTED IS            2
NOW ENTER:  C ,TO CONFIRM
C

SECTION             1
 TYPE OF PRESCRIBED BC ?
1<<    du/dn=0 & dv/dn=0
2<<    h  & du/dn=0 & dv/dn=0
3<<    mx & dv/dn=0
4<<    my & du/dn=0
5<<    h  & mx & dv/dn=0
6<<    h  & my & du/dn=0
7<<    mx & my
8<<    h  & mx & my
9<<    u  & dv/dn=0
10<<   du/dn=0 & v
11<<   u  & v
12<<   mx & v
13<<   u  & my
14<<   h & my/mx
15<<   my/mx
          8
```

```
THE TYPE SELECTED IS              8
NOW ENTER:  C ,TO CONFIRM
        C

FIRST
EXTREME NODE OF SECTION                1
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          1
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
          1
ELEMENT              1 NODE              1
LOCAL NODE NUMBER:              1
ROW             1 COLUMN             1
NOW ENTER:  C ,TO CONFIRM
        C

SECOND
EXTREME NODE OF SECTION                1
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          1
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
          1
ELEMENT              1 NODE              1
LOCAL NODE NUMBER:              1
ROW             1 COLUMN             1
NOW ENTER:  C ,TO CONFIRM
        C

SECTION             2
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
        5
THE TYPE SELECTED IS              5
NOW ENTER:  C ,TO CONFIRM
C

SECTION             2
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   h  & du/dn=0 & dv/dn=0
3<<   mx & dv/dn=0
4<<   my & du/dn=0
5<<   h  & mx & dv/dn=0
```

```
6<<   h  & my & du/dn=0
7<<   mx & my
8<<   h  & mx & my
9<<   u  & dv/dn=0
10<<  du/dn=0 & v
11<<  u  & v
12<<  mx & v
13<<  u  & my
14<<  h & my/mx
15<<  my/mx
        4
THE TYPE SELECTED IS             4
NOW ENTER:  C ,TO CONFIRM
       C

FIRST
EXTREME NODE OF SECTION               2
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          1
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
          2
ELEMENT                 1 NODE            9
LOCAL NODE NUMBER:                2
ROW              1 COLUMN             1
NOW ENTER:  C ,TO CONFIRM
       C

SECOND
EXTREME NODE OF SECTION               2
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          1
ELEMENT COLUMN ?
         19
LOCAL NODE NUMBER ?
          1
ELEMENT               127 NODE          145
LOCAL NODE NUMBER:                1
ROW              1 COLUMN            19
NOW ENTER:  C ,TO CONFIRM
       C

SECTION               3
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
          3
```

```
THE TYPE SELECTED IS           3
NOW ENTER:  C ,TO CONFIRM
C
DOES THIS OUTLET FACE THE AXIAL DIRECTION ?

1<< YES
2<< NO
          1

SECTION               3
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   h  & du/dn=0 & dv/dn=0
3<<   mx & dv/dn=0
4<<   my & du/dn=0
5<<   h  & mx & dv/dn=0
6<<   h  & my & du/dn=0
7<<   mx & my
8<<   h  & mx & my
9<<   u  & dv/dn=0
10<<  du/dn=0 & v
11<<  u  & v
12<<  mx & v
13<<  u  & my
14<<  h & my/mx
15<<  my/mx
          2
THE TYPE SELECTED IS           2
NOW ENTER:  C ,TO CONFIRM
        C

FIRST
EXTREME NODE OF SECTION           3
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          1
ELEMENT COLUMN ?
         19
LOCAL NODE NUMBER ?
          2
ELEMENT           127 NODE          153
LOCAL NODE NUMBER:            2
ROW             1 COLUMN          19
NOW ENTER:  C ,TO CONFIRM
        C

SECOND
EXTREME NODE OF SECTION           3
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          1
ELEMENT COLUMN ?
        119
```

```
LOCAL NODE NUMBER ?
         2
ELEMENT            827 NODE            0
LOCAL NODE NUMBER:                2
ROW               1 COLUMN          119
NOW ENTER:  C ,TO CONFIRM
        C

SECTION               4
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
         4
THE TYPE SELECTED IS           4
NOW ENTER:  C ,TO CONFIRM
C
DOES THIS OUTLET FACE THE AXIAL DIRECTION ?

1<< YES
2<< NO
          1

SECTION               4
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   h  & du/dn=0 & dv/dn=0
3<<   mx & dv/dn=0
4<<   my & du/dn=0
5<<   h  & mx & dv/dn=0
6<<   h  & my & du/dn=0
7<<   mx & my
8<<   h  & mx & my
9<<   u  & dv/dn=0
10<<  du/dn=0 & v
11<<  u  & v
12<<  mx & v
13<<  u  & my
14<<  h & my/mx
15<<  my/mx
         2
THE TYPE SELECTED IS           2
NOW ENTER:  C ,TO CONFIRM
        C

FIRST
EXTREME NODE OF SECTION           4
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          1
ELEMENT COLUMN ?
         19
```

```
LOCAL NODE NUMBER ?
         3
ELEMENT            127 NODE          154
LOCAL NODE NUMBER:                 3
ROW              1 COLUMN           19
NOW ENTER:  C ,TO CONFIRM
        C

SECOND
EXTREME NODE OF SECTION              4
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
         7
ELEMENT COLUMN ?
         19
LOCAL NODE NUMBER ?
         2
ELEMENT            133 NODE          159
LOCAL NODE NUMBER:                 2
ROW              7 COLUMN           19
NOW ENTER:  C ,TO CONFIRM
        C

SECTION              5
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
         3
THE TYPE SELECTED IS            3
NOW ENTER:  C ,TO CONFIRM
C
DOES THIS OUTLET FACE THE AXIAL DIRECTION ?

1<< YES
2<< NO
         1

SECTION              5
 TYPE OF PRESCRIBED BC ?
1<<    du/dn=0 & dv/dn=0
2<<    h  & du/dn=0 & dv/dn=0
3<<    mx & dv/dn=0
4<<    my & du/dn=0
5<<    h  & mx & dv/dn=0
6<<    h  & my & du/dn=0
7<<    mx & my
8<<    h  & mx & my
9<<    u  & dv/dn=0
10<<   du/dn=0 & v
11<<   u  & v
12<<   mx & v
```

```
13<<  u  & my
14<<  h & my/mx
15<<  my/mx
          2
THE TYPE SELECTED IS                    2
NOW ENTER:  C ,TO CONFIRM
        C

FIRST
EXTREME NODE OF SECTION                      5
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
           7
ELEMENT COLUMN ?
         19
LOCAL NODE NUMBER ?
           3
ELEMENT              133 NODE            160
LOCAL NODE NUMBER:                    3
ROW                 7 COLUMN            19
NOW ENTER:  C ,TO CONFIRM
        C

SECOND
EXTREME NODE OF SECTION                      5
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
           7
ELEMENT COLUMN ?
         19
LOCAL NODE NUMBER ?
           3
ELEMENT              133 NODE            160
LOCAL NODE NUMBER:                    3
ROW                 7 COLUMN            19
NOW ENTER:  C ,TO CONFIRM
        C

SECTION                  6
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL     BOUNDARY
          5
THE TYPE SELECTED IS                    5
NOW ENTER:  C ,TO CONFIRM
C

SECTION                  6
 TYPE OF PRESCRIBED BC ?
1<<    du/dn=0 & dv/dn=0
```

```
2<<   h  & du/dn=0 & dv/dn=0
3<<   mx & dv/dn=0
4<<   my & du/dn=0
5<<   h  & mx & dv/dn=0
6<<   h  & my & du/dn=0
7<<   mx & my
8<<   h  & mx & my
9<<   u  & dv/dn=0
10<<  du/dn=0 & v
11<<  u  & v
12<<  mx & v
13<<  u  & my
14<<  h & my/mx
15<<  my/mx
          4
THE TYPE SELECTED IS            4
NOW ENTER:  C ,TO CONFIRM
        C

FIRST
EXTREME NODE OF SECTION              6
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          7
ELEMENT COLUMN ?
         19
LOCAL NODE NUMBER ?
          4
ELEMENT              133 NODE           152
LOCAL NODE NUMBER:                4
ROW               7 COLUMN             19
NOW ENTER:  C ,TO CONFIRM
        C

SECOND
EXTREME NODE OF SECTION              6
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          7
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
          3
ELEMENT                7 NODE           16
LOCAL NODE NUMBER:                3
ROW               7 COLUMN              1
NOW ENTER:  C ,TO CONFIRM
        C

SECTION              7
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
```

```
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
          2
THE TYPE SELECTED IS            2
NOW ENTER:  C ,TO CONFIRM
C

SECTION             7
 TYPE OF PRESCRIBED BC ?
1<<    du/dn=0 & dv/dn=0
2<<    h  & du/dn=0 & dv/dn=0
3<<    mx & dv/dn=0
4<<    my & du/dn=0
5<<    h  & mx & dv/dn=0
6<<    h  & my & du/dn=0
7<<    mx & my
8<<    h  & mx & my
9<<    u  & dv/dn=0
10<<  du/dn=0 & v
11<<  u  & v
12<<  mx & v
13<<  u  & my
14<<  h & my/mx
15<<  my/mx
          8
THE TYPE SELECTED IS            8
NOW ENTER:  C ,TO CONFIRM
         C

FIRST
EXTREME NODE OF SECTION              7
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          7
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
          4
ELEMENT             7 NODE              8
LOCAL NODE NUMBER:              4
ROW             7 COLUMN            1
NOW ENTER:  C ,TO CONFIRM
         C

SECOND
EXTREME NODE OF SECTION              7
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          7
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
```

```
          4
ELEMENT              7 NODE             8
LOCAL NODE NUMBER:               4
ROW               7 COLUMN              1
NOW ENTER:  C ,TO CONFIRM
         C

SECTION              8
 TYPE OF BOUNDARY ?
1<< INFLOW  BOUNDARY
2<< INFLOW  BOUNDARY CORNER
3<< OUTFLOW BOUNDARY CORNER
4<< OUTFLOW BOUNDARY
5<< WALL    BOUNDARY
           1
THE TYPE SELECTED IS              1
NOW ENTER:  C ,TO CONFIRM
C

SECTION              8
 TYPE OF PRESCRIBED BC ?
1<<   du/dn=0 & dv/dn=0
2<<   h  & du/dn=0 & dv/dn=0
3<<   mx & dv/dn=0
4<<   my & du/dn=0
5<<   h  & mx & dv/dn=0
6<<   h  & my & du/dn=0
7<<   mx & my
8<<   h  & mx & my
9<<   u  & dv/dn=0
10<<  du/dn=0 & v
11<<  u  & v
12<<  mx & v
13<<  u  & my
14<<  h & my/mx
15<<  my/mx
           8
THE TYPE SELECTED IS              8
NOW ENTER:  C ,TO CONFIRM
         C

FIRST
EXTREME NODE OF SECTION              8
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          7
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
          1
ELEMENT              7 NODE             7
LOCAL NODE NUMBER:               1
ROW               7 COLUMN              1
NOW ENTER:  C ,TO CONFIRM
```

```
        C

SECOND
EXTREME NODE OF SECTION                8
NODE LOCATION
ELEMENT CO-ORDINATES
ELEMENT ROW ?
          1
ELEMENT COLUMN ?
          1
LOCAL NODE NUMBER ?
          4
ELEMENT                1 NODE                2
LOCAL NODE NUMBER:                4
ROW              1 COLUMN                1
NOW ENTER:  C ,TO CONFIRM
         C

THE BOUNDARY-CONDITION CODE FILE HAS BEEN
CREATED
```

APPENDIX G

A Tutorial to JOEGRAPH

```
PROCEDURES AVAILABLE:
1<< SURFACE PROJECTED ON A PLANE (MESH)
2<< SURFACE
3<< VELOCITY FIELD PROJECTED ON A PLANE
4<< VELOCITY FIELD ON SURFACE
5<< CONTOUR PLOT PROJECTED ON A PLANE
6<< CONTOUR PLOT PROJECTED ON A PLANE WITH
            AREA SHADING
7<< CONTOUR PLOT ON SURFACE
8<< CONTOUR PLOT ON SURFACE WITH
            AREA SHADING

PLEASE ENTER THE NUMBER OF PROCEDURES YOU
WISH TO HAVE CARRIED OUT SIMULTANEOUSLY.
3

PLEASE ENTER THE NUMBER OF
 HEADING LINES (L<=4)
1

PLEASE ENTER HEADING LINE #               1
Joegraph Plot

PLEASE ENTER THE VERTICAL DISTANCE
(IN INCHES) BETWEEN PLOTS.(>0)
0.1

1<< SURFACE PROJECTED ON A PLANE (MESH)
2<< SURFACE
3<< VELOCITY FIELD PROJECTED ON A PLANE
4<< VELOCITY FIELD ON SURFACE
5<< CONTOUR PLOT PROJECTED ON A PLANE
6<< CONTOUR PLOT PROJECTED ON A PLANE WITH
        AREA SHADING
7<< CONTOUR PLOT ON SURFACE
8<< CONTOUR PLOT ON SURFACE WITH
        AREA SHADING

PLEASE ENTER THE CODE FOR TASK NUMBER:             1
2

PLEASE ENTER THE INITIAL OF THIS
TASK'S INDEPENDENT VARIABLE.
h

h
WHICH VARIABLE WOULD YOU LIKE TO PLOT ?
```

```
1 << FREE SURFACE HEIGHT.
2 << X-COMPONENT VELOCITY.
3 << Y-COMPONENT VELOCITY.
4 << TEMPERATURE/CONCENTRATION.
1

DO WISH TO HAVE A COLOR SCALE
PLOTTED FOR THIS VARIABLE ?
(FOR COLOR PLOTS ONLY)
1 <<  YES
0 <<   NO
0

1<< SURFACE PROJECTED ON A PLANE  (MESH)
2<< SURFACE
3<< VELOCITY FIELD PROJECTED ON A PLANE
4<< VELOCITY FIELD ON SURFACE
5<< CONTOUR PLOT PROJECTED ON A PLANE
6<< CONTOUR PLOT PROJECTED ON A PLANE WITH
       AREA SHADING
7<< CONTOUR PLOT ON SURFACE
8<< CONTOUR PLOT ON SURFACE WITH
       AREA SHADING

PLEASE ENTER THE CODE FOR TASK NUMBER:            2
3

PLEASE ENTER THE INITIAL OF THIS
TASK'S INDEPENDENT VARIABLE.
V
V

 DO YOU PREFER COLOR VECTORS ?
 1<< YES
 0<<  NO
1

WOULD YOU LIKE A SCALE PLOTTED FOR THIS?
 1<< YES
 0<<  NO
1

WOULD YOU LIKE TO SPECIFY CUSTOM
VECTOR VIEWING PARAMETERS ?
0 << NO
1 << YES

1

ENTER VELOCITY-VECTOR MAGNIFICATION
 CONSTANT
0.5

ENTER VECTOR WIDTH-TO-LENGTH  0-5
1
```

```
ENTER VECTOR SIZE 0-6
1

ENTER COLUMN STRIDE
2

ENTER ROW STRIDE
1

1<< SURFACE PROJECTED ON A PLANE  (MESH)
2<< SURFACE
3<< VELOCITY FIELD PROJECTED ON A PLANE
4<< VELOCITY FIELD ON SURFACE
5<< CONTOUR PLOT PROJECTED ON A PLANE
6<< CONTOUR PLOT PROJECTED ON A PLANE WITH
       AREA SHADING
7<< CONTOUR PLOT ON SURFACE
8<< CONTOUR PLOT ON SURFACE WITH
       AREA SHADING

PLEASE ENTER THE CODE FOR TASK NUMBER:              3
8

PLEASE ENTER THE INITIAL OF THIS
TASK'S INDEPENDENT VARIABLE.
U

U
WHICH VARIABLE WOULD YOU LIKE TO PLOT ?
1 << FREE SURFACE HEIGHT.
2 << X-COMPONENT VELOCITY.
3 << Y-COMPONENT VELOCITY.
2

FOR CONTOUR PLOT NUMBER:                1
MIN=   -1.331216      MAX=    4.564607
DO YOU WISH TO SPECIFY THE
CONTOUR LEVELS BY DESIGNATING
THEIR UPPER AND LOWER LIMITS,
ALONG WITH THE # OF CONTOURS ?
0 <<  NO
1 << YES
1

ENTER ZMIN
0
ENTER ZMAX
1.

ENTER # OF LEVELS  (L>=2)
18

DO YOU NEED TO CUSTOMIZE
THE APPEARANCE OF THE GRAPH ?
0
```

APPENDIX H

Performance Analysis of FEMFRS

For the multiple spur dike problem involving a horizontal bed and narrow outlet, the following figure depicts the percentages of CPU time spent in each subroutine of FEMFRS. The matrix algebra problem (solved in the subroutine SOLVR) requires only 3.8% of the total execution time.

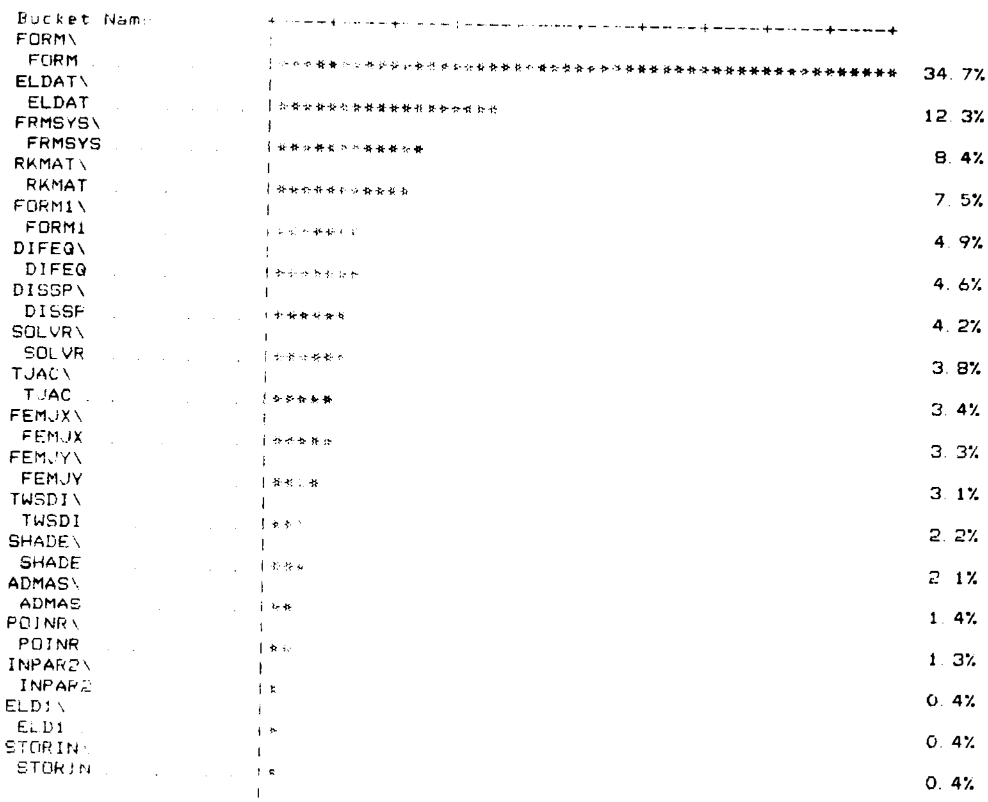

Figure A - 3. Performance analysis of the matrix solving subroutine.

APPENDIX I

Determination of Optimum values for β

The most effective level of artificial dissipation was determined in the following manner. The three β coefficients are initially set to zero. The solution is run for a fixed number of small time steps. The free surface depth solution is recorded in an IMAGEN file. The number of blocks of storage is noted. β_1 is incremented by some appropriate amount (perhaps 0.05) and the simulation run from the first initial condition. This procedure is repeated, incrementing β_1 to the point where the file size reaches a local minimum. The value of β_1 at the local minimum is considered to be optimal. β_1 and β_3 are then set to zero while β_2 is varied in similar manner to determine the optimum value for β_2. The procedure is repeated for β_3.

This procedure proved entirely effective for β_1, β_2 and β_3 during the mentioned test. The explanation for this method is as follows. At low levels of β, the numerical dispersion error creates "wiggles" in the solution which necessitate storage in the form of excess contour lines. As β is increased the solution passes the smoothest (fewest number of wiggles) solution at the optimum damping level. Any increase beyond this point deteriorates the solution increasing the wiggles, and eventually (quickly) leading to divergence. As divergence is approached the file size actually decreases as the contour lines are restricted to [typically] one "spike" located somewhere in the computational domain.

The optimum values of the study were found to be β = (0.70, 0.33, 0.40). Unfortunately, when used in combination, these provide too much dispersive control (i.e., a wiggly solution). However, the three optimum parameters may be used to determine an optimum ratio of β.

VITA

Albert McGuire Hines was born to Martha Jane Tucker Hines and the late Richard Harwell Hines on April 16, 1960 in Montgomery, Alabama. At age five, he moved to Columbia, Tennessee where he attended McDowell Elementary School, Whitthorne Junior High School and Central High School. His graduation from high school was in June of 1978.

In Fall of the 1978, he first attended The University of Tennessee in the college of engineering. The most important event of his life occurred January 1, 1980 when he met the true God, Christ Jesus. After that time, not only his grades steadily improved, but he was given a genuine purpose in life – to know God and help build His kingdom on Earth.

In June of 1982, after receiving his B.S. degree in mechanical engineering, he enrolled in the M.S. program in mechanical engineering. After changing majors twice, he found a thesis project in the Engineering Science and Mechanics department both interesting and suitable.

He was married to Leslie Beatrice Martin on August 6, 1988 in Cades Cove Methodist Church, in the Great Smoky Mountains. Her companionship, commitment and encouragement have been remarkable and wonderful. Albert resides presently at 160 Taliwa Court in Knoxville, Tennessee.