HYDRAULICS BRANCH
OFFICIAL FILE COPY

Bureau of Reclamation
HYDRAULICS BRANCH
OFFICE
FILE COPY
WHEN BORROWED RETURN PROMPTLY

NUMERICAL MODEL OF LANDSLIDE-GENERATED WAVES

PAP 456

PREPARED FOR

U.S. BUREAU OF RECLAMATION
DENVER, COLORADO

CONTRACT NO. 0-07-81-S0134

PREPARED BY:

Wen-Li Chiang
David Divoky
Peter Parnicky
Wayne Wier

TETRA TECH REPORT T-3427

NOVEMBER, 1981

PAP 456

TETRA TECH, INC.
630 NORTH ROSEMEAD BLVD.
PASADENA, CALIFORNIA 91107

NOTICE

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, or any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for any third party's use, or the results of such use, of any information, apparatus product or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights.

NUMERICAL MODEL OF LANDSLIDE-GENERATED WAVES

PREPARED FOR

U.S. BUREAU OF RECLAMATION
DENVER, COLORADO

CONTRACT NO. 0-07-81-S0134

PREPARED BY:

Wen-Li Chiang
David Divoky
Peter Parnicky
Wayne Wier

TETRA TECH REPORT T-3427

NOVEMBER, 1981

TETRA TECH, INC.
630 NORTH ROSEMEAD BLVD.
PASADENA, CALIFORNIA 91107

7-2090 (4-81)
Water and Power

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO.	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
4. TITLE AND SUBTITLE Numerical Model Landslide-Generated Waves		5. REPORT DATE November 1981 6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) Wen-Li Chiang, David Divrky, Peter Parnicky, Wayne Weir		8. PERFORMING ORGANIZATION REPORT NO. T-3427
9. PERFORMING ORGANIZATION NAME AND ADDRESS Tetra Tech. Inc. 630 North Rosemead Blvd. Pasadena, California 91107		10. WORK UNIT NO. 11. CONTRACT OR GRANT NO. 0-07-81-S0134
12. SPONSORING AGENCY NAME AND ADDRESS Bureau of Reclamation PO Box 25007, Engineering and Research Center Denver Federal Center Denver, CO 80225		13. TYPE OF REPORT AND PERIOD COVERED 14. SPONSORING AGENCY CODE DIBR

15. SUPPLEMENTARY NOTES
Uses data from REC-ERC-82-9
Hydraulic Model Studies of Landslide-Generated Water Waves - Morrow Point Reservoir

16. ABSTRACT

A computer program has been developed for the propagation of waves throughout a reservoir using finite difference methods. Waves may be generated within the code by using one of two landslide descriptions or a fault displacement under the reservoir. The two landslide descriptions allow the code to generate velocity-time relationships or the user to input slide strips with individual velocity time relationships.

Verification of the computer code is done by comparison with two physical model studies and one historical case history.

The report contains the theory used, a user's guide, examples and a listing.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS--	Water Waves Landslides Fault Displacement	Numerical Analysis Computer Program
b. IDENTIFIERS--	Morrow Point Dam Morrow Point Reservoir	Gunnison River Colorado
c. COSATI Field/Group	COWRR:	SRIM:

18. DISTRIBUTION STATEMENT	19. SECURITY CLASS (THIS REPORT)	21. NO. OF PAGES
Available from the National Technical Information Service, Operations Division, 5285 Port Royal Road, Springfield, Virginia 22161.	UNCLASSIFIED	
	20. SECURITY CLASS (THIS PAGE) UNCLASSIFIED	22. PRICE

GPO 831-316

ACKNOWLEDGEMENTS

This study was performed within the Engineering Division of Tetra Tech under the direction of Mr. David Divoky. Dr. Wen-Li Chiang was Principal Investigator. He was assisted by Messrs. Wayne Wier and Peter Parnicky. Additional support was provided by Drs. Frank Wu and Michael Chen and Messrs. Al Robertson and Shahab Nazarie.

The development of the model described here benefited greatly from the close participation of Messrs. David Harris and Clifford Pugh of the U.S. Bureau of Reclamation who defined the model requirements, provided laboratory data for test comparisons and offered detailed comments, criticisms, and suggestions throughout the study. The comments and suggestions of Dr. Ted Yang of the Bureau of Reclamation are also gratefully acknowledged.

TABLE OF CONTENTS

PART A NUMERICAL MODEL

Section		Page
A1.0	INTRODUCTION	1
A2.0	MATHEMATICAL MODEL	7
	A2.1 Hydrodynamic Submodel	7
	A2.1.1 Partial Differential Equations	7
	A2.1.2 Numerical Method	15
	A2.2 Wave-generating Submodels	20
	A2.2.1 "Forced" Landslide	21
	A2.2.2 "Natural" Landslide	22
	A2.2.3 Bottom Motion Generated Waves	25
A3.0	NUMERICAL EXPERIMENTS	26
	A3.1 Morrow Point Reservoir, Colorado	26
	A3.1.1 Description	26
	A3.1.2 Simulation	29
	A3.2 Lituya Bay, Alaska	43
	A3.2.1 Description	43
	A3.2.2 Simulation	43
	A3.3 Mica Reservoir--Example of an Earth Dam	49
	A3.3.1 Description	49
	A3.3.2 Simulation	50
	A3.3.3 Additional remarks	59
	A3.4 Downstream Flood Routing	60
A4.0	CONCLUSIONS AND RECOMMENDATIONS	66
APPENDIX A1	LIST OF NOTATIONS	67

TABLE OF CONTENTS (Cont'd)

PART B USER'S GUIDE

Section		Page
B1.0	INTRODUCTION	70
B2.0	HYDRODYNAMIC MODEL	72
	B2.1 General Description	72
	B2.2 Spacings and Time Step	75
	B2.3 Input Data for Water Depth	79
	B2.4 Boundaries	80
	B2.4.1 Dam	81
	B2.4.2 Open Boundary	84
	B2.4.3 Partially Reflective Boundary	84
	B2.4.4 Totally Reflective Boundary	89
	B2.5 Discharge Coefficient	90
	B2.6 Other Parameters	91
B3.0	"FORCED" LANDSLIDE"	94
B4.0	"NATURAL" LANDSLIDE	107
B5.0	BOTTOM MOTION GENERATED WAVES	110
B6.0	DESCRIPTION OF INPUT PARAMETERS	112
B7.0	OUTLINE OF OUTPUT DATA	121
B8.0	PROGRAM LISTING	123
B9.0	SAMPLE INPUT AND OUTPUT	232
	B9.1 Listing of Sample Input Data	232
	B9.2 Description of Input Data	235
	B9.3 Listing of Sample Output	239
APPENDIX B1	SUMMARY OF MODIFICATIONS TO DAMBRK	256
LIST OF REFERENCES		259

LIST OF FIGURES

Figure		Page
A1	Definition sketch of coordinate system and notation	8
A2	Graphical representation of the total transmission open boundary condition	13
A3	Effects of energy dissipation coefficient in the case of wave oscillation in a closed tank	16
A4	Staggered scheme	18
A5	Site map for Morrow Point Reservoir, Colorado	27
A6	Schematic diagram of model landslide	28
A7	Grid network for Morrow Point Reservoir numerical model	30
A8	Horizontal velocities of landslides in both physical and numerical models	31
A9	Surface fluctuations at Probe 3, due to landslide A at Morrow Point Reservoir	32
A10	Surface fluctuations at Probe 14, due to landslide A at Morrow Point Resevoir	33
A11	Surface fluctuations at Probe 3, due to landslide B at Morrow Point Reservoir	34
A12	Surface fluctuations at Probe 10, due to landslide B at Morrow Point Reservoir	35
A13	Surface fluctuations at Probe 12, due to landslide B at Morrow Point Reservoir	36
A14	Surface fluctuations at Probe 3, due to landslide C at Morrow Point Reservoir	37
A15	Surface fluctuations at Probe 10, due to landslide C at Morrow Point Reservoir	38
A16	Surface fluctuations at Probe 14, due to landslide C at Morrow Point Reservoir	39
A17	Computed discharge hydrograph at Morrow Point Dam	42
A18	Lituya Bay	44
A19	Upper reach of Lituya Bay showing approximate location of the Gilbert Inlet slide	45

LIST OF FIGURES (Cont'd)

Figure		Page
A20	Map showing locations of nine stations for which computed wave elvations are showing in the accompanying table.....	48
A21	Configuration of Mica Reservoir near Mica Dam showing position of Little Chief Ridge landslide and wave gage stations 3, 7, 9, and 10.....	51
A22	Geometry of Little Chief Ridge landslide in lab series 2.....	53
A23	Measured and computed elevations of the first wave at Stations 3 and 7.....	55
A24	Measured and computed elevations of the first wave at Stations 9 and 10.....	56
B1	Finite-difference representation of a reservoir with the grid spacing equal to, from top to bottom, 500 feet, 250 feet, and 125 feet.....	73
B2	Finite-difference representation of a reservoir, with the elevation points falling on intersection of the grid system.....	74
B3	Definition sketch of space-staggered scheme.....	76
B4	Definition sketch for the location of dams....	82
B5	Definition sketch of various kinds of boundaries.....	85
B6	Relationship between the conventional transmission coefficient and the energy dissipation coefficient used in this model for a particular case of a simple wave oscillation in a 1-D basin.....	87
B7	Dependence of SA121 peak height at Morrow Point Dam on energy dissipation coefficient...	88
B8	Discharge coefficient for vertical-faced ogee crest.....	92
B9	Definition sketch of landslide input data.....	95
B10	Definition sketch of landslide.....	98
B11	Effects of the number of sub-divisions on the peak discharge over the dam, total overtopping, and peak wave height at the center of the dam, for the case of a landslide with a vertical front surface.....	100
B12	Maximum error of submerged cross-sectional area for a chosen number of sub-spacings, MSUB	102

LIST OF FIGURES (Cont'd)

Figure		Page
B13	The measured drag on unit axial length of a circular cylinder (A=2a), a sphere ($A=\pi a^2$), and on a circular disk normal to the stream ($A=\pi a^2$), all of radius a.....	104
B14	Average viscous drag coefficient, CDV, for sand-roughened plates.....	105
B15	Definition sketch of fault-induced bottom motion.....	111
B16	Sensitivity test of the discharge coefficient in the Weir formula.....	238

LIST OF TABLES

Table		Page
A1	Computed maximum offshore wave elevations, Lituya Bay	49
B1	Example input data to describe dams in Figure B4(b)	83
B2	Drag coefficients	104
B3	Specific gravity and unit weight of various rocks	108
B4	Index of input variables	120

PART A
NUMERICAL MODEL

A1.0 INTRODUCTION

The objective of this study was to develop a numerical model to simulate the generation, propagation, and dissipation of rockfall or landslide-generated water waves in bays, lakes, reservoirs, fjords, and rivers. Such waves are a major cause for concern in the vicinity of dams which may be overtopped and damaged, with flooding downstream which may cause the loss of life. Other areas of concern include damage to intake structures, spillways, power houses, and shoreline structures.

It is possible for a large land mass to become loose and plunge into the nearby water body under certain conditions such as earthquake, bomb explosion, heavy rain (especially after fire), and so on. When a large landslide intrudes into a large body of water at a high speed, a huge water wave can be generated. Large landslide-generated waves have been observed in Kenai Lake, Disenchantment Bay, and Lituya Bay, Alaska; Vajont Reservoir, Italy; Shimabara Bay, Japan; Yanahuin Lake, Peru; and many Norwegian fjords and lakes (Jorstad, 1968; Miller, 1960; McCulloch, 1966; Noda, 1970; Raney and Butler, 1975; Slingerland and Voight, 1979). The combined death toll from these events exceeds twenty thousand people.

The process of a landslide entering water is very complicated. The factors affecting the characteristics of landslide-generated waves include the dimensions, velocity, orientation, and initial location of the landslide, and the dimensions of the water body. The slide velocity depends on local slope, frictional resistance in both the air and water, and the shape and dimensions of the landslide mass.

Some assumptions have to be adopted to build a practical, predictive model.

Two kinds of landslide submodels have been constructed in this study. For a "forced" landslide, the motion of the landslide is predetermined by measurement, calculation, or some other engineering method. For a "natural" landslide, the landslide is considered as a train of segments moving down a variable slope under a combination of different natural forces. All segments are assumed to move at the same time-varying horizontal velocity so the slide does not deform.

To describe the generation and propagation of landslide-generated waves, an analytical approach (e.g., Kennard, 1949; Kranzer and Keller, 1958; Noda, 1969; Madsen, 1970) has little practical application because the geometry has to be simple and the water depth has to be constant. Several physical models have been built to study the landslide-generated waves. Wiegel (1955), Kuba (1968), Madsen (1970), Wiegel, et al. (1970), and Das and Wiegel (1971) used a one-dimensional flume and made some laboratory experiments to study the characteristics of landslide-generated waves. Kamphuis and Bowering (1970) derived empirical expressions for wave height and period as functions of distance from the point of entry, based on dimensional analysis and experimental data obtained in a one-dimensional flume. Hydraulic models have been conducted to investigate landslide-generated waves in Mica Reservoir (Western Canada Hydraulic Laboratories, 1970), Lake Koocanusa and Libby Dam (Davidson and Whalin, 1974; Davidson and McCartney, 1975), Revelstoke Reservoir (Northwest Hydraulic Consultants Ltd., 1976), and Morrow Point Reservoir (Pugh and Harris, 1982).

Laboratory models, while often necessary, are both expensive and cumbersome, and do not permit rapid, efficient con-

sideration of alternate conditions. A numerical model , on the other hand, offers great convenience and flexibility. The landslide mechanism, wave propagation in a reservoir of varied depth, overtopping, and forces on the dam can all be computed by utilizing a digital computer. If there is a danger of overtopping or even a dam-break, the landslide-wave model can be coupled with a flood routing model to study the potential downstream hazards. After a reasonably generalized numerical model is constructed, it is easy to adapt it for other physical conditions or other study areas.

Noda (1969, 1970) provided numerical results for the cases of a vertically falling box and a horizontal piston. The initial water depth is constant. In the case of variable water depth, laws of conservation of mass and momentum are used to simulate the wave motions. Tetra Tech (1975) applied the shallow water equations, neglecting the nonlinear advective terms, to simulate the wave propagation due to a potential landslide from Mt. Baker into Baker Lake, Washington. Koutitas (1977) used a finite element method to predict landslide-generated waves in a one-dimensional reservoir. Raney and Butler (1975) applied the shallow water equations, including advective terms, to simulate two-dimensional wave propagation. Numerical results were compared to experimental data from a hydraulic model of Libby Dam and Lake Koocanusa. Slingerland and Voight (1979) showed comparisons among alternate descriptions of large water waves generated by landslides.

For Downie Slide into Revelstoke Reservoir, Mercer, et al. (1979) used a hybrid model which consisted of three submodels. The first submodel was a physical model to simulate the landslide motion and the propagation and attenuation of the landslide-generated waves. A time series of wave height was measured seven miles downstream from the landslide site. At this point, the surface fluctuation is already smoothed

and the wave period is elongated. The measured wave heights were then fed into the second submodel which was a numerical model of a one-dimensional wave propagating downstream to about 1.3 miles upstream of Revelstoke Dam. The runup on Revelstoke Dam was then found from the third submodel, a physical model, based on the computed wave data at the point 1.3 miles upstream of the dam.

In the present study, the nonlinear, depth-averaged continuity and momentum equations are used to simulate the propagation and dissipation of water waves in a variable-depth basin. Vertical accelerations are neglected. The major forcing mechanism of the landslide-generated wave is incorporated into the continuity equation. When a landslide mass intrudes into the water, the displaced water volume raises the water surface and the increased elevation-gradient induces water motions. The secondary mechanism--drag induced water motion--is incorporated into the momentum equations.

An explicit finite-difference method is used to solve the partial differential equations. A mapping technique (Chiang, 1979) is used such that those points outside of the computational domain do not consume extra storage space. About one third to one half of the array dimensions can be eliminated in this way. Four types of boundaries are allowed in the model: dam, totally reflective boundary, totally transmissive boundary, and partially reflective boundary. The model is very flexible. Effects of wave reflection, refraction, and shoaling are all implicit in the model. Energy dissipation caused by wave breaking in nearshore areas is represented by a partial reflective boundary.

The complete model has been exercised by simulating several laboratordy and historical slides. These simulations include waves due to three potential landslides in the vicinity of Morrow Point Dam, Colorado, landslides near Mica

Dam (an earth dam in British Columbia), and the Lituya Bay rockslide which occurred on July 9, 1958. All results are compared to either laboratory measurements or field observations.

At the completion of a simulation with this model, the discharge hydrograph over the dam (if any) is stored into a file. The file is then used to supply the upstream hydrograph for a flood routing model.

A major emphasis in the development of this model was user convenience. For example, if certain variables are not specified, the program adopts average default values. In particular, the program is extremely flexible in the ways which a landslide may be described. The "forced landslide" submodel requires only very simple input but permits description of an extremely broad variety of motions. Similarly, the "natural landslide" submodel eliminates the necessity for the user to specify any slide motions at all, while requiring only very simple input. In both modes, the user can perform parametric studies--varying, say, slide path angle, friction slope, and the like--by changing a single input value.

A similar effort has been expended to make the printed output as useful as possible. A large number of output options are controlled by the user allowing him to expand or contract the volume of output data at will. The output includes documentation of all input data and liberal use of descriptive graphics which permit the user to see at a glance that data has been entered correctly. Line printer plots of computed time histories and hydrographs are also available if requested by the user.

This report consists of two parts. Part A describes the numerical model and the numerical experiments. Part B is the Uner's Guide for the complete model, LSWAVE.

In Part A, Chapter A2 presents the basic concepts and mathematical models used in the study. The model consists of a hydrodynamical submodel and three wave-generating submodels. Numerical methods used in solving the governing equations are described.

Chapter A3 depicts the procedures and results of several numerical experiments using LSWAVE. The experiments include those for landslide-generated waves in Morrow Point Reservoir, Lituya Bay, and Mica Reservoir. The results of each simulation are discussed. The transition of the hydrograph data from LSWAVE to a downstream flood routing model is described, and difficulties in downstream flood routing are discussed.

Chapter A4 gives a summary of the conclusions and recommendations. Appendix A1 contains a list of notations used in Part A of this report. The list of references is merged together with the references for Part B, and is found at the very end of the report.

A2.0 MATHEMATICAL MODEL

The model consists of a hydrodynamic submodel and three wave-generating submodels. The hydrodynamic submodel is used to simulate the propagation and dissipation of the generated waves. The main wave-generating mechanism considered in this study is a landslide, although a simple dip-slip earthquake displacement is also included.

A right-handed cartesian (x-y-z) coordinate system fixed on the earth with the z-axis vertically upwards is used in this report (Figure A1). The x-y plane rests on the initially still water surface of the basin. The water surface fluctuates about this datum plane. The depth is defined as the negative elevation, relative to this datum plane, of the bottom surface. That is, the depth is measured positive downward from the still water surface. Water velocities mentioned in this report are all depth-averged velocities.

A2.1 Hydrodynamic Submodel

A2.1.1 Partial differential equations

Assuming hydrostatic pressure and neglecting the vertical motion, a set of depth-averaged equations is derived from the continuity equation and the Navier-Stokes equations:

$$\frac{\partial \eta}{\partial t} + \frac{\partial (HU)}{\partial x} + \frac{\partial (HV)}{\partial y} = -\frac{\partial D}{\partial t} \tag{1}$$

$$\frac{\partial U}{\partial t} + U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} + g\frac{\partial \eta}{\partial x} = -F_x \tag{2}$$

$$\frac{\partial V}{\partial t} + U\frac{\partial V}{\partial x} + V\frac{\partial V}{\partial y} + g\frac{\partial \eta}{\partial y} = -F_y \tag{3}$$

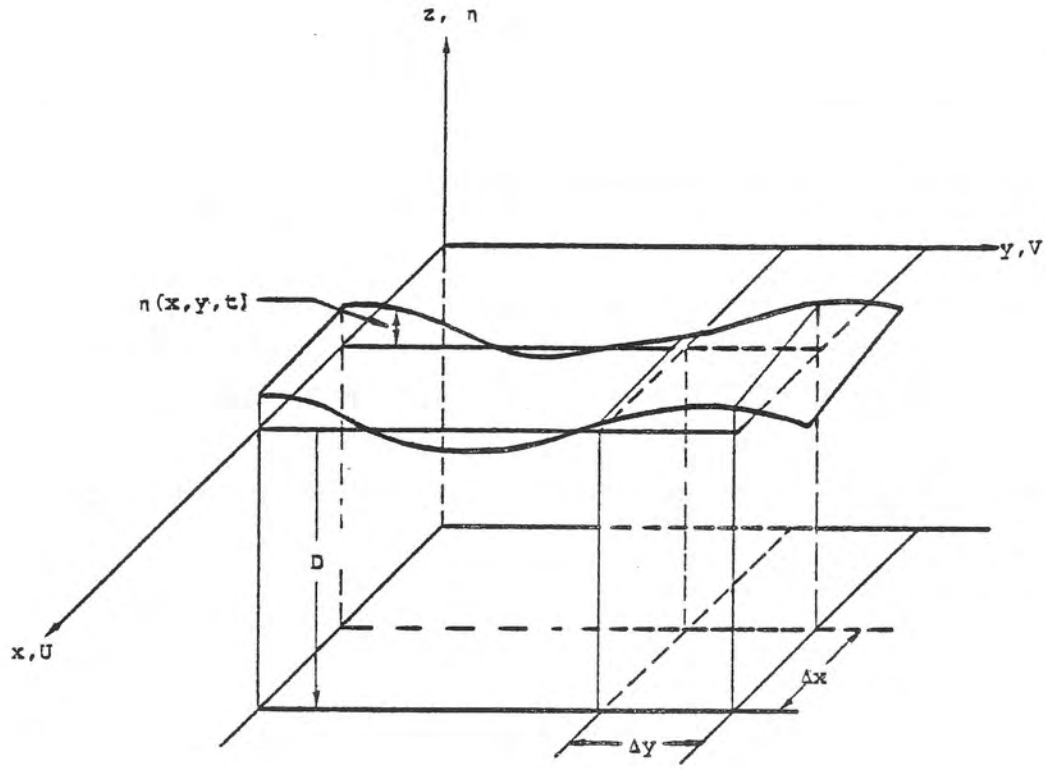


FIGURE A1: Definition sketch of coordinate system and notation.

where D = water depth relative to initial water surface;

F_x, F_y = total friction along x- and y-directions, respectively;

g = gravitational acceleration;

$H = \eta + D$, total water depth relative to instantaneous water surface;

t = time;

u, v = water velocities along x- and y-directions, respectively;

x, y = coordinates;

and η = surface elevation relative to initial water surface (positive upward).

Since this model is to be applied to a relatively small area, the Coriolis effect is neglected and the curvature of the Earth is not considered.

Terms on the right hand sides of Eqs. 1 to 3 are forcing functions. The variable D is the water depth relative to the initial water surface. The negative of this value is the bottom elevation with respect to the intial water surface. Therefore, Eq. 1 shows the conservation of mass by considering the rate of change of bottom elevation caused by the intrusion of a landslide.

The terms on the right hand sides of the momentum equations 2 and 3 include effects of bottom friction and drag due to a landslide. Surface stress and turbulent friction are not considered in this study. At a location outside of the landslide path, the resistance force is equal to the usual bottom stress.

The x-component of the bottom stress can be expressed as (see Chiang, 1979)

$$S_{bx} = \frac{1}{C_c^{\,2}} \rho g U (U^2 + V^2)^{\frac{1}{2}} \tag{4}$$

where ρ is the unit mass of water, and C_c is the Chezy coefficient, which is related to Manning's coefficient, n, by (see Henderson, 1966);

$$C_c = \frac{1}{n}R^{1/6} \tag{5}$$

where R is the hydraulic radius. In a two-dimensional flow, R can be approximated by the height of the water column, H, and therefore:

$$C_c = \frac{1}{n}H^{1/6} \tag{6}$$

When Manning's n is expressed in FPS units instead of the MKS system, the above expression should be:

$$C_c = \frac{1.486}{n}H^{1/6} \tag{7}$$

Combining Eqs. 4 to 7, the x-component of resistance in Eq. 2 is expressed as

$$F_x = \frac{S_{bx}}{\rho H} = \frac{n^2 gU(U^2+V^2)^{\frac{1}{2}}}{1.486^2\, H^{4/3}} \tag{8}$$

Similarly, the y-component resistance is expressed as

$$F_y = \frac{S_{by}}{\rho H} = \frac{n^2 gV(U^2+V^2)^{\frac{1}{2}}}{1.486^2\, H^{4/3}} \tag{9}$$

At a location covered by the landslide, the resistance comes from the drag induced by the slide. The drag is normally expressed as (Schlichting, 1960; Rosenhead, 1963; Batchelor, 1967)

$$\text{Drag} = \frac{1}{2}\rho C_D A V_r^2 \tag{10}$$

where A is some kind of cross-sectional area of the submerged body, C_D is called the drag coefficient, and V_r is

the relative velocity of the slide. This equation is assumed to be applicable to an unsteady state when the instantaneous velocity is used for V_r. The drag can be considered as a combination of pressure drag (called form drag), D_p, and viscous drag (called surface drag), D_v. At a location covered by the landslide, the bottom stress is replaced by the drag, and the total horizontal resistance is

$$F = -\frac{D_p+D_v}{\rho H A_s}\cos\alpha$$

$$= -\frac{1}{2}\frac{V_r^2}{HA_s}(C_{Dp}A_p+C_{Dv}A_v)\cos\alpha \qquad (11)$$

where α is the angle between the instantaneous slide direction and the horizontal plane, C_{Dp} and C_{Dv} respectively are the pressure drag coefficient and the viscous drag coefficient, and A_p, A_v, and A_s are appropriate cross-sectional areas. In this study, A_p is defined as the projected area of the submerged part of the slide, on a plane normal to the direction of motion; A_v is the projected area of the submerged slide on a plane paralllel to the local bottom slope; and A_s is the horizontal area of projection of the submerged part of the slide. The far-field velocity is assumed to be zero, and the relative velocity, V_r, is taken to be the slide velocity.

The basic structure of the hydrodynamic model is Eqs. 1 to 3, with F defined either by Eqs. 8 and 9, or by Eq. 11, depending on whether the specific location is covered by the moving landslide or not.

There are four kinds of boundaries in this model:

(a) Numerical truncation boundaries (open boundaries) beyond which computations are, for reasons of economy, not continued (primarily upstream) but at which some approximations must be imposed;

(b) The dam face which acts as a finite height reflective wall which allows overtopping only for sufficiently large waves; otherwise, the wave is totally reflected back;

(c) Solid boundaries which reflect waves without transmission or dissipation; and

(d) Partially reflective boundaries on which waves may, to some degree, break and be dissipated.

The treatment of truncation boundaries is straight-forward. The condition of complete wave transmission is derived from the one-dimensional continuity equation

$$\frac{\partial \eta}{\partial t} + \frac{\partial (HU)}{\partial x} = 0 \tag{12}$$

In a long wave,

$$U \simeq C\frac{\eta}{H} \tag{13}$$

where $$C \simeq (gH)^{\frac{1}{2}} \tag{14}$$

is the wave celerity. Then, from Eq. 12, (Hwang and Divoky, 1970),

$$\begin{aligned} \frac{\partial \eta}{\partial t} &\simeq -\frac{\partial (C\eta)}{\partial x} \\ &\simeq -C\frac{\partial \eta}{\partial x} \end{aligned} \tag{15}$$

A schematic representation of the treatment is depicted in Figure A2 in which superscripts n and n+1 denote time levels while subscripts b and i denote the boundary point and the neighboring interior point, respectively.

At a dam, a weir formula applies:

$$\frac{\partial \eta}{\partial t} = -C\frac{\partial \eta}{\partial x}$$

$$\eta_b^{n+1} = \eta_b^n + C\frac{\Delta t}{\Delta x}(\eta_i^n - \eta_b^n)$$

$$C = (gH)^{\frac{1}{2}} = [g(D+\eta)]^{\frac{1}{2}}$$

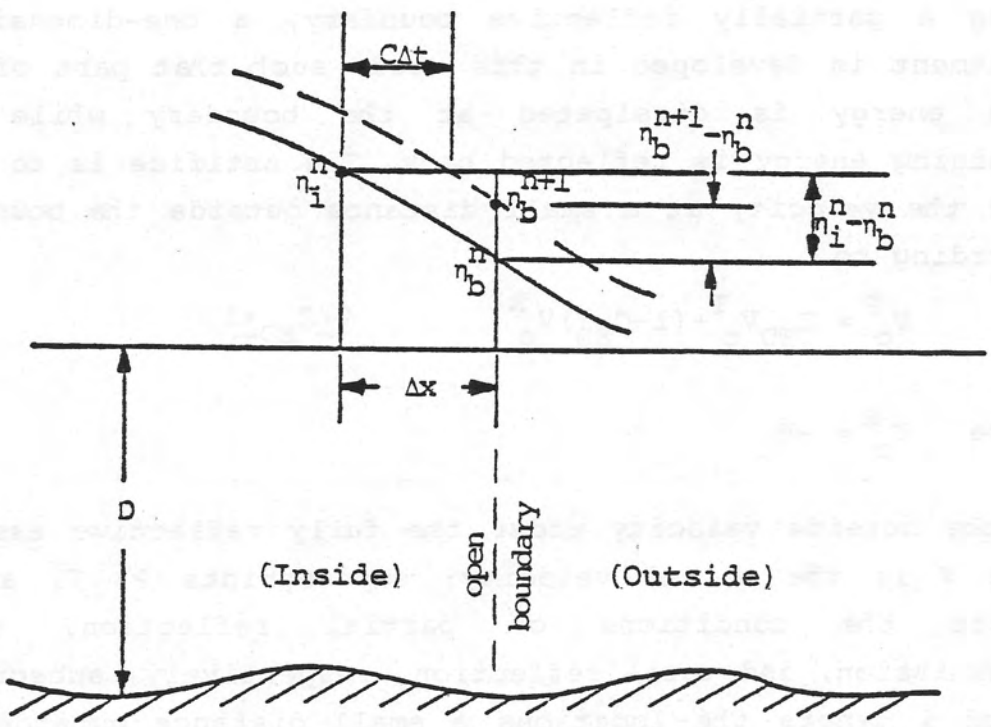


FIGURE A2: Graphical representation of the total transmission open boundary condition. (After Hwang and Divoky, 1975).

$$q = C_d' H_t (2gH_t)^{\frac{1}{2}}$$

$$= C_d H_t^{3/2} \tag{16}$$

where $C_d = C_d' (2g)^{\frac{1}{2}}$ (17)

is the discharge coefficient used in this model, q is the discharge over a unit width of the dam, C_d is a dimensionless discharge coefficient, and H_t is the total head above the dam.

A solid boundary is a reflective boundary across which the the normal velocity is zero. It is obtained, in this model, by setting an abnormally high dam along the boundary such that q is always zero.

Along a partially reflective boundary, a one-dimensional treatment is developed in this study such that part of the wave energy is dissipated at the boundary while the remaining energy is reflected back. The artifice is to compute the velocity at a small distance outside the boundary according to

$$V_o^P = C_{ED} V_o^T + (1-C_{ED}) V_o^R \qquad 0 \leq C_{ED} \leq 1 \tag{18}$$

where $V_o^R = -V_i$ (19)

is the outside velocity under the fully reflective assumption V is the normal velocity; superscripts P, T, and R denote the conditions of partial reflection, total transmission, and total reflection, respectively; subscripts o and i denote the locations a small distance outside and the same small distance inside the boundary; and C_{ED} is here termed the "energy dissipation coefficient." Eq. 19 is obtained by assuming the boundary to be a perfectly symmetric reflective boundary. The outside velocity under the fully transmissive assumption V_o^T, is obtained by the combination of Eq. 15 the mass conservative equation.

The energy dissipation coefficient, C_{ED}, although indirectly proportional to the commonly used "transmission coefficient" has no direct physical meaning. The relationship between C_{ED} and the transmission coefficient depends on the characteristics of the wave under study, such as wave period, and on some parameters of the numerical model such as grid spacing, computational time step.

In order to illustrate the effect of C_{ED}, a simple test was performed for a one-dimensional standing wave. The basin has a constant depth of 50 meters and a length of one half wave length which was set at 1680 meters. Partial reflective boundaries were imposed at both ends of the basin. The grid spacing was 70 meters and the computational time step was 3 seconds. The initial profile was sinusoidal. The time histories of wave elevation for the cases of C_{ED} equal to 1.0, 0.7, 0.3, and 0, are shown in Figure A3. The solid line and the dotted line represent the fluctuations of the surface elevation at the two ends of the basin. When C_{ED} = 1.0, the initial wave is totally transmitted after a half cycle. When C_{ED} = 0, the wave oscillates inside the closed basin without dissipation. In the other two intermediate cases, the results show different rates of wave decay. The relation between C_{ED} and the estimated transmitted coefficient in this case is plotted in Figure B6 of Part B. This test shows that although C_{ED} is not a physical variable, its use effectively simulates energy dissipation.

A2.1.2 Numerical method

In general, Eqs. 1 to 3 cannot be solved analytically. A finite difference method of numerical integration is adopted here.

The typical time duration of landslides in this study is about 10 seconds. The computational time step should therefore be about 1 second or less, in order to have good reso-

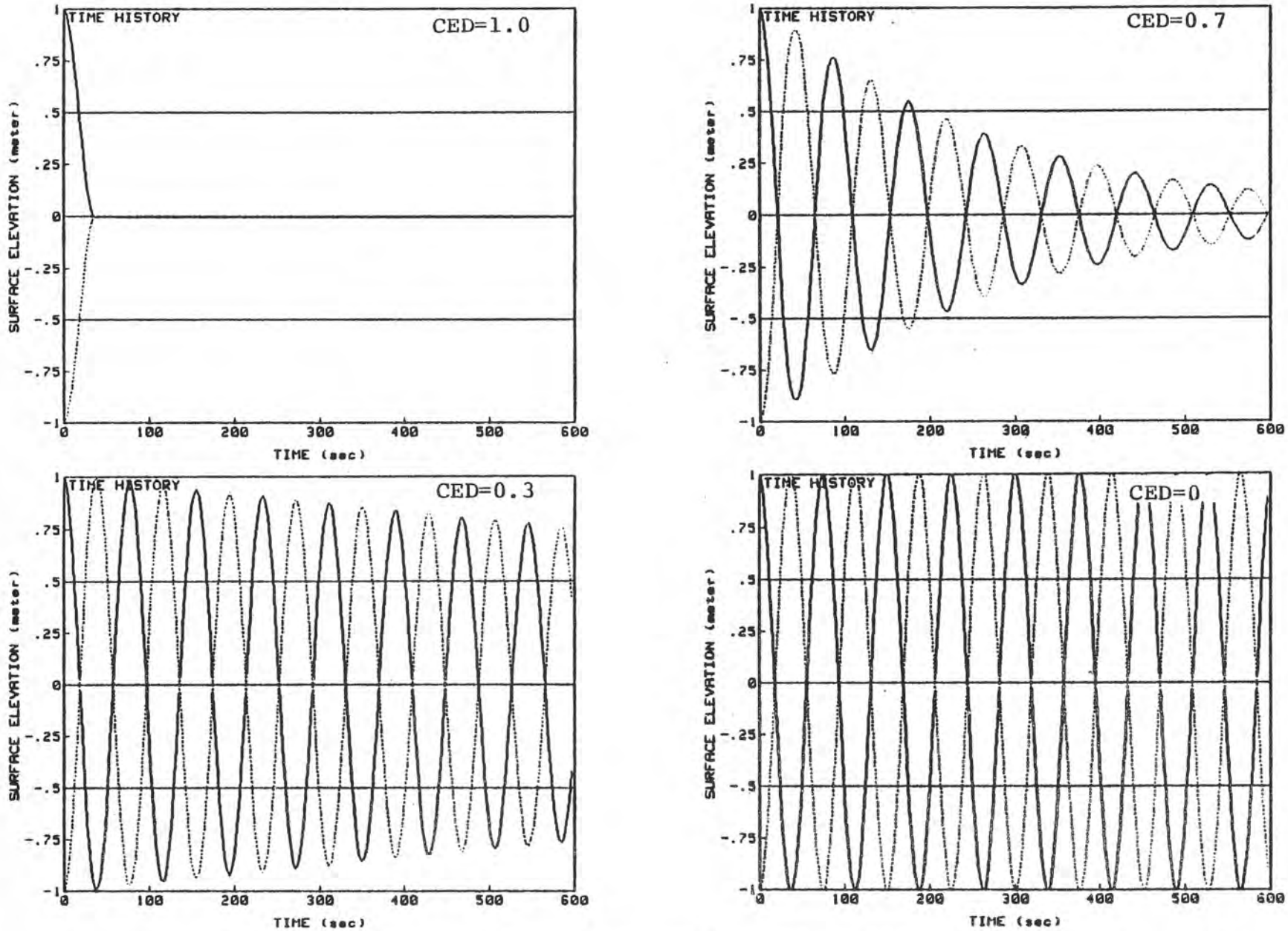


FIGURE A3: Effects of energy dissipation coefficient in the case of wave oscillation in a closed tank.

lution of the disturbance. For a typical water depth of about 400 feet, and a grid spacing of 200 feet or larger, the associated Courant number, which is a function of time step, grid spacing, and wave celerity, would then be lower than unity. Thus, there is little advantage to consider an implicit scheme and a straightforward explicit scheme is used in this study in order to minimize the computational effort.

A central-difference formulation is used in transforming the partial differential equations to finite difference equations. A space-staggered scheme is adopted in this study to increase the efficiency of programming. In this scheme, surface elevation, η, and two components of velocity, U and V, are described at different locations of a grid (see Figure A4). Depth, D, is defined at the same point as η. Furthermore, the scheme is time-staggered. That is, the surface elevations are defined at time levels $n-\frac{1}{2}$, $n+\frac{1}{2}$, n+3/2, etc., while velocities are defined at time levels n, n+1, n+2, etc., as can be seen in the following finite difference equations.

The finite difference equations at a point (i,j,n), where i and j are indices of the spatial coordinates, and n represents the time step, are derived from Eqs. 1 to 3 as:

$$\frac{1}{\Delta t}\left(\eta_{i,j}^{n+\frac{1}{2}}-\eta_{i,j}^{n-\frac{1}{2}}\right)$$

$$+\frac{1}{2\Delta x}\left[\left(H_{i+1,j}^{n-\frac{1}{2}}+H_{i,j}^{n-\frac{1}{2}}\right)U_{i+\frac{1}{2},j}^{n}-\left(H_{i,j}^{n-\frac{1}{2}}+H_{i-1,j}^{n-\frac{1}{2}}\right)U_{i-\frac{1}{2},j}^{n}\right]$$

$$+\frac{1}{2\Delta y}\left[\left(H_{i,j+1}^{n-\frac{1}{2}}+H_{i,j}^{n-\frac{1}{2}}\right)V_{i,j+\frac{1}{2}}^{n}-\left(H_{i,j}^{n-\frac{1}{2}}+H_{i,j-1}^{n-\frac{1}{2}}\right)V_{i,j-\frac{1}{2}}^{n}\right]$$

$$=-\frac{1}{\Delta t}\left(D_{i,j}^{n+\frac{1}{2}}-D_{i,j}^{n-\frac{1}{2}}\right) \qquad (20)$$

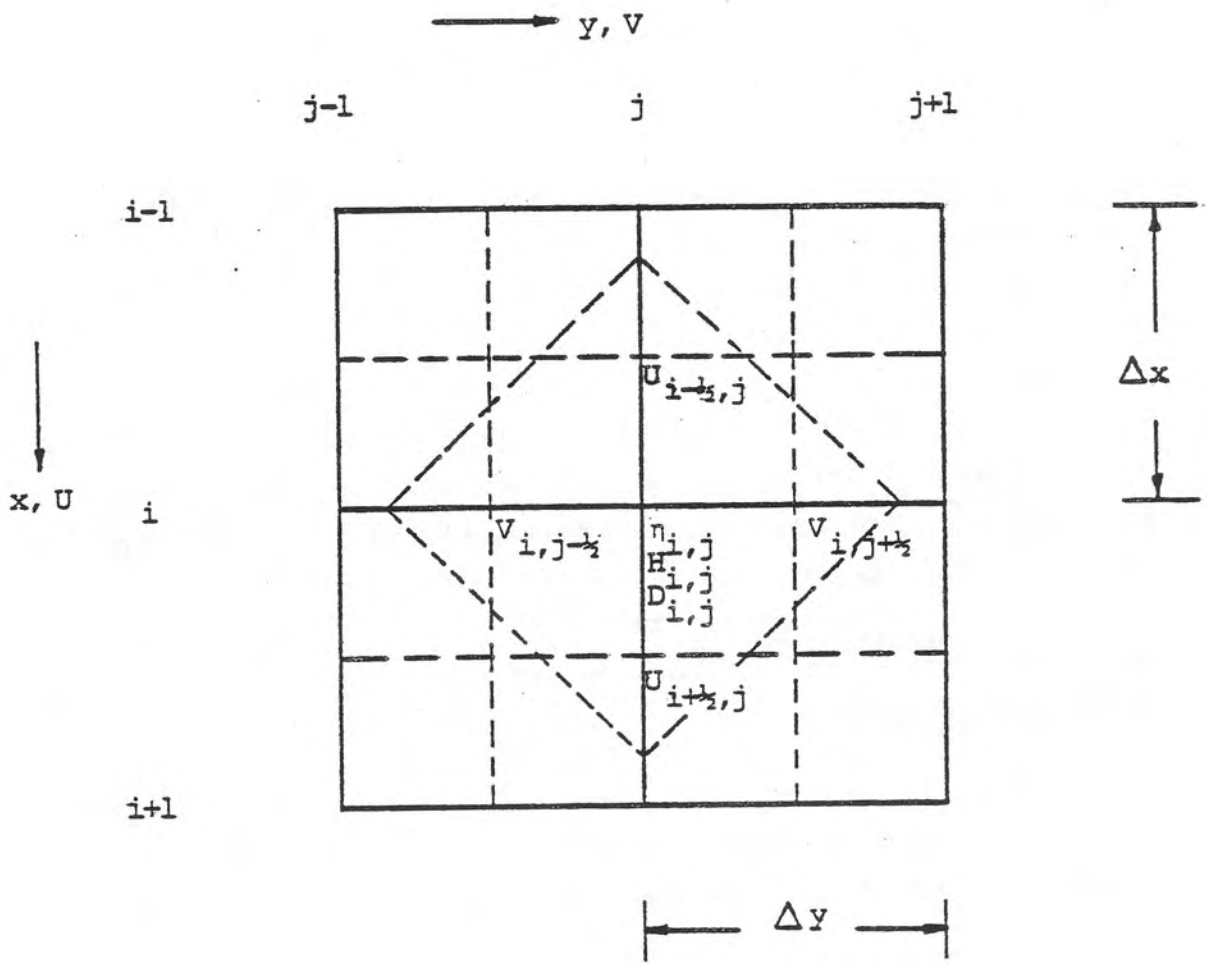


FIGURE A4: Staggered scheme. (After Brandsma, et al., 1974).

$$\frac{1}{\Delta t}\left(U^{n+1}_{i+\frac{1}{2},j}-U^{n}_{i+\frac{1}{2},j}\right) + \frac{1}{2\Delta x}U^{n}_{i+\frac{1}{2},j}\left(U^{n}_{i+3/2,j}-U^{n}_{i-\frac{1}{2},j}\right)$$

$$+ \frac{1}{2\Delta y}\overline{V}^{n}_{i+\frac{1}{2},j}\left(U^{n}_{i+\frac{1}{2},j+1}-U^{n}_{i+\frac{1}{2},j-1}\right) + \frac{g}{\Delta x}\left(\eta^{n+\frac{1}{2}}_{i+1,j}-\eta^{n+\frac{1}{2}}_{i,j}\right)$$

$$= -(F_x)^{n+\frac{1}{2}}_{i+\frac{1}{2},j} \tag{21}$$

and

$$\frac{1}{\Delta t}\left(V^{n+1}_{i,j+\frac{1}{2}}-V^{n}_{i,j+\frac{1}{2}}\right) + \frac{1}{2\Delta x}\overline{U}^{n}_{i,j+\frac{1}{2}}\left(V^{n}_{i+1,j+\frac{1}{2}}-V^{n}_{i-1,j+\frac{1}{2}}\right)$$

$$+ \frac{1}{2\Delta y}V^{n}_{i,j+\frac{1}{2}}\left(V^{n}_{i+3/2}-V^{n}_{i,j-\frac{1}{2}}\right) + \frac{g}{\Delta y}\left(\eta^{n+\frac{1}{2}}_{i,j+1}-\eta^{n+\frac{1}{2}}_{i,j}\right)$$

$$= -(F_y)^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} \tag{22}$$

where $\overline{U}_{i,j+\frac{1}{2}} \equiv \frac{1}{4}\left(U_{i-\frac{1}{2},j}+U_{i-\frac{1}{2},j+1}+U_{i+\frac{1}{2},j}+U_{i+\frac{1}{2},j+1}\right)$ (23)

$$\overline{V}_{i+\frac{1}{2},j} \equiv \frac{1}{4}\left(V_{i,j-\frac{1}{2}}+V_{i,j+\frac{1}{2}}+V_{i+1,j-\frac{1}{2}}+V_{i+1,j+\frac{1}{2}}\right) \tag{24}$$

and F_x, following Eqs. 8 and 11, is expressed as

$$(F_x)^{n+\frac{1}{2}}_{i+\frac{1}{2},j}$$

$$= \delta \frac{C'}{\left(H^{n+\frac{1}{2}}_{i+1,j}\right)^{3/4} + \left(H^{n+\frac{1}{2}}_{i,j}\right)^{3/4}} U^n_{i+\frac{1}{2},j} \left[\left(U^n_{i+\frac{1}{2},j}\right)^2 + \left(\overline{V}^n_{i+\frac{1}{2},j}\right)^2\right]^{\frac{1}{2}}$$

$$- (1-\delta)\frac{1}{2} \frac{\left[(V_r)^{n+\frac{1}{2}} \cos\theta\right]^2}{(\Delta x)(\Delta y)}$$

$$\left[\frac{\gamma C_{Dp}(A_p)^{n+\frac{1}{2}} + C_{Dv}(A_v)^{n+\frac{1}{2}}_{i+1,j}}{H^{n+\frac{1}{2}}_{i+1,j}} + \frac{\gamma C_{Dp}(A_p)^{n+\frac{1}{2}} + C_{Dv}(A_v)^{n+\frac{1}{2}}_{i,j}}{H^{n+\frac{1}{2}}_{i,j}}\right] \cos\alpha \tag{25}$$

in which δ is either 0 when the landslide is passing the point $(i+\frac{1}{2}, j)$ or 1 otherwise; γ is 1 only when its associated term is applied to the first element of the landslide, otherwise, 0; θ is the forward direction of the slide, relative to the x-axis; and the constant C' is

$$C' \equiv \frac{2n^2 g}{1.486^2} \tag{26}$$

The diamond shape in Figure A4 indicates the numerical domain of dependence for Eq. 20, the continuity equation.

Eqs. 21 and 22 are actually applied at $(i+\frac{1}{2}, j)$ and $(i, j+\frac{1}{2})$, respectively. In the computer coding, integers are used for indices and $i+\frac{1}{2}$ and $j+\frac{1}{2}$ are denoted by i and j, respectively. The index of time step $n+\frac{1}{2}$ is denoted by n+1. Thus the U's and V's in Figure A4 are coded as U(i-1,j), U(i,j), V(i,j-1), and V(i,j) although they are defined at points half way between elevation points which have exact integer indices (i,j).

A2.2 Wave-generating Submodels

Three submodels are incorporated to simulate the wave-

generating forces. The basic concepts are described in the following three sections.

A2.2.1 "Forced" landslide

The "forced" landslide submodel is used to simulate the generation of waves due to a landslide for which the velocity history is pre-determined. The motion of the model landslide is forced to follow the time history of the velocity specified by the user. The input velocity can be one measured in the laboratory or one computed or estimated in any manner. The slide geometry can be quite complex and deformable.

The controlling factors of the "forced" landslide include:

(a) Slide dimensions, determining the volume of water displaced and partly determining the horizontal length scales associated with the waves;

(b) Slide velocity, determining the rapidity with which fluid is displaced, governing the initial wave steepness and energy;

(c) Slide orientation, affecting the directionality of the initial wave system; and

(d) Slide location, influencing subsequent propagation patterns and intensity.

From the starting position, direction, and the velocity, the model determines the spatial distribution of the slide at any instant. The change of bottom elevation at a computational grid point due to the intrusion of the landslide can then be calculated from one time step to the next. The rate of change is inserted on the right hand side of Eq. 20. This change rate of bottom elevation is the primary forcing function of the landslide-generated waves. When multiplied by the grid area, $(\Delta x)(\Delta y)$, this rate becomes the rate of intruding volume of the landslide.

At a grid covered by the landslide, the resistance terms in Eqs. 21 and 22, F_x and F_y, are contributed by the drag due

to the landslide and not by the bottom friction. The drag is a secondary forcing mechanism in generating waves.

A grid system different from the main grid used in the hydrodynamic submodel is used to describe the distribution of the landslide (Figure B8 in Part B). Three dimensional features of a landslide can be easily reproduced. Across the landslide, the width is equally divided into a number of slices. Each slice has its own specification of shape and its own velocity history. All slices are taken to move in the same direction. Along the landslide, the total length is divided into several segments with unequal horizontal spacing. The thickness of the sliding mass is specified independently at each segment for each slice. When the slide moves along an uneven slope, each segment is assumed to preserve its own height and horizontal spacing and all segments in a given slice move at a uniform speed during a time step. Since each slice has its own shape and velocity history, the simulated motions can be extremely general.

A2.2.2 "Natural" landslide

A "natural" landslide submodel computes the time history of the slide velocity based on the natural characteristics of the landslide and the topography. The velocity of water is neglected and the time history of the slide velocity is computed prior to the wave simulation.

Based on the starting position and direction of landslide motion, the potential path of the slide is established. The acceleration and velocity of the slide are computed as functions of position along this path at horizontal steps Δs which is the smaller of Δx and Δy. The total length of the landslide is divided into several segments of equal length Δs. The heights (thickness) of these segments are computed by interpolation using the input data as described for the forced landslide, and the several forces acting on

these segments are summed to determine the total force on the slide.

The equation of motion is:

$$\frac{a}{\cos\alpha}(\rho_s V_s + K\rho V_s') = \rho_s g V_s(\sin\alpha)$$

$$- (\rho_s V_s - \rho V_s')g(\cos\alpha)(\tan\phi)$$

$$- \rho g V_s'(\sin\alpha)$$

$$- \tfrac{1}{2}\rho C_{Dp} V_r^2 A_p$$

$$- \tfrac{1}{2}\rho C_{Dv}\left(\frac{V_r}{\cos\alpha}\right)^2 A_v \qquad (27)$$

in which

a = horizontal acceleration of landslide;

$a/\cos\alpha$ = acceleration along the sliding slope;

A_p = projected area of the submerged part of landslide, on a plane normal to the direction of slide;

A_v = projected area of the submerged part of landslide, on a plane parallel to the local bottom surface;

C_{Dp} = pressure drag coefficient;

C_{Dv} = viscous drag coefficient;

K = added mass coefficient;

V_s = total volume of landslide;

V_s = submerged volume of landslide;

α = inclination (angle between the instantaneous sliding direction and the horizontal plane);

ρ = unit mass of water;

ρ_s = unit mass of landslide;

ϕ = friction angle.

The left hand side of the equation is the total force accelerating the slide mass and the added hydrodynamic mass. The five terms on the right hand side are gravitational force, friction, buoyancy, pressure drag, and viscous drag, respectively. The equation is actually describing the horizontal motion of the center of mass of the landslide segment.

When a finite number of landslide segment is considered, each segment has its own volume and slope. Eq. 27 is rewritten as

$$a\left[\rho_s \sum_i (V_s)_i + K\rho \sum_i (V_s')_i\right] = \rho_s g \sum_i (V_s \sin\alpha \cos\alpha)_i$$

$$- g\tan\phi \sum_i \left[(\rho_s V_s - \rho V_s')(\cos^2\alpha)\right]_i$$

$$- \rho g \sum_i (V_s' \sin\alpha \cos\alpha)_i$$

$$- \frac{1}{2}\rho C_{Dp} V_r^2 W (HSL)_{max}$$

$$- \frac{1}{2}\rho C_{Dv} W \sum_i \left[\left(\frac{V_r}{\cos\alpha}\right)^2 \frac{\Delta s}{\cos\alpha} \cos\alpha\right]_i \qquad (28)$$

where W is the width of the slide and $(HSL)_{max}$ denotes the maximum submerged height of the slide. The horizontal acceleration, a, in Eq. 28 is an averaged accleration which represents the average horizontal motion of the entire slide.

The horizontal velocity is computed by

$$V_h^{n+1} = V_h^n + \frac{1}{2}(\Delta t)^{n+\frac{1}{2}}(a^{n+1} + a^n) \qquad (29)$$

where the time interval, Δt, for the slide to advance a fixed distance s is obtained from

$$\Delta s = V_h^n (\Delta t)^{n+\frac{1}{2}} + \frac{1}{4}(a^n + a^{n+1}) \left[(\Delta t)^{n+\frac{1}{2}}\right]^2 \tag{30}$$

With the initial velocity set to zero, the velocity as a function of time can be found using Eqs. 25 to 27. When this time history of velocity is available, the "forced" landslide routines can be used to continue the computation of waves generated by the "natural" landslide.

A2.2.3 Bottom Motion Generated Waves

A small submodel is included in this study to simulate the waves induced by a fault motion. A straight fault line passes through the water body and the bottom to one side of the fault line is either raised or lowered instantaneously. The surface elevation changes simultaneously according to the conservation of mass. The wave is produced by the gradient of surface elevation and the subsequent flow motions.

A3.0 NUMERICAL EXPERIMENTS

A3.1 Morrow Point Reservoir, Colorado

A3.1.1 Description

Morrow Point Dam, completed in 1963, is a double curvature thin-arch concrete dam. It is 740 ft long and 469 ft high. Gunnison River upstream of the dam makes Morrow Point Reservoir in Gunnison County and Montrose County, Colorado. In the first three miles of the reach upstream from Morrow Point Dam, there are three potential landslide areas (Pugh and Harris, 1982) which are designated as "A", "B", and "C", as shown in Figure A5, in this study. Slide A, located approximately 2,500 ft upstream from the right abutment of the dam, appears to have the greatest threat to the dam (Safety of Dams Evaluation Team for Morrow Point Dam, 1979).

A physical model was built at the Bureau of Reclamation Hydraulic Laboratory to study the characteristics of landslide-generated waves in Morrow Point Reservoir (Pugh and Harris, 1982). The landslide was represented by a block with a shape of a reversed wedge, sliding down a slope of 29° into the reservoir as shown in Figure A6. The widths of the block were, in prototype, 1250 ft for slide A and 1000 ft for both slide B and slide C. The blocks were not submerged when they came to a stop at the end of the slopes.

The model was constructed to an undistorted scale of 1:250. Since both the time ratio and velocity ratio are square roots of the length ratio, they are 1:15.81. With these ratios, the dimensional analysis for Eqs. 1 to 3 shows that the model is similar to the prototype as long as the ratio of n, Manning's friction coefficient, is 1:2.51.

In the laboratory, water level measurements were made at 13 locations throughout the reservoir. Probes are numbered 2 to

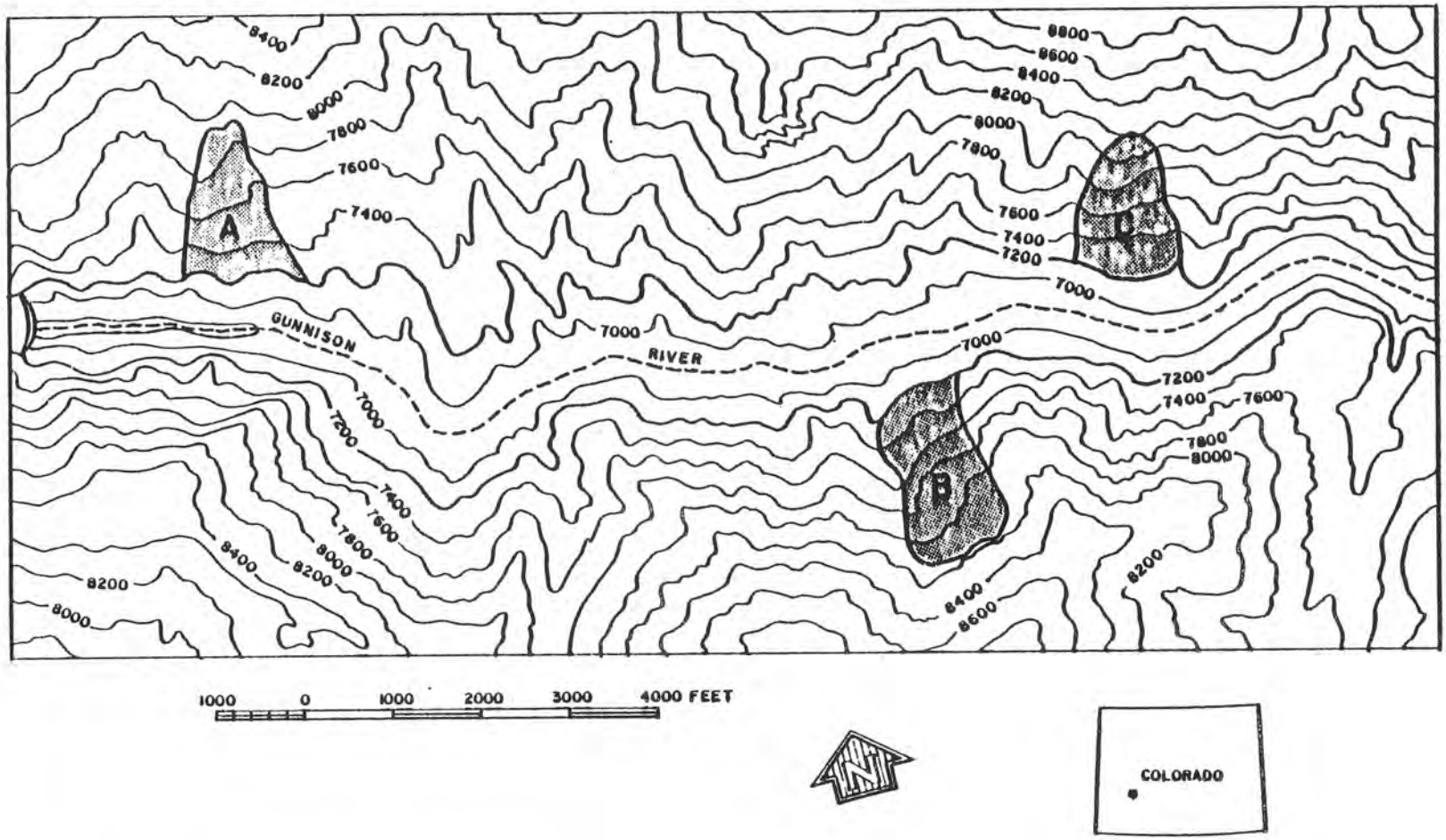
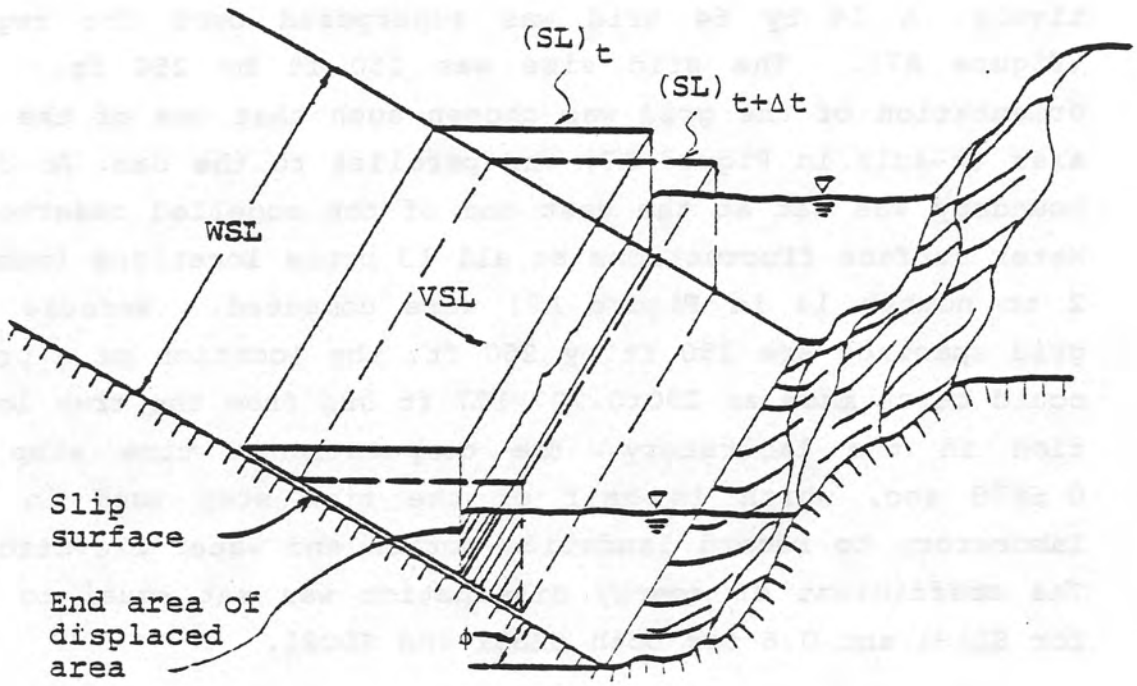


FIGURE A5: Site map for Morrow Point Reservoir, Colorado.



LEGEND:

$(SL)_t$	-- Landslide position at t
$(SL)_{t+\Delta t}$	-- Landslide position at t+Δt
VSL	-- Landslide velocity
WSL	-- Landslide width
ϕ	-- Angle of slope

FIGURE A6: Schematic diagram of model landslide. (After Pugh and Harris, 1982).

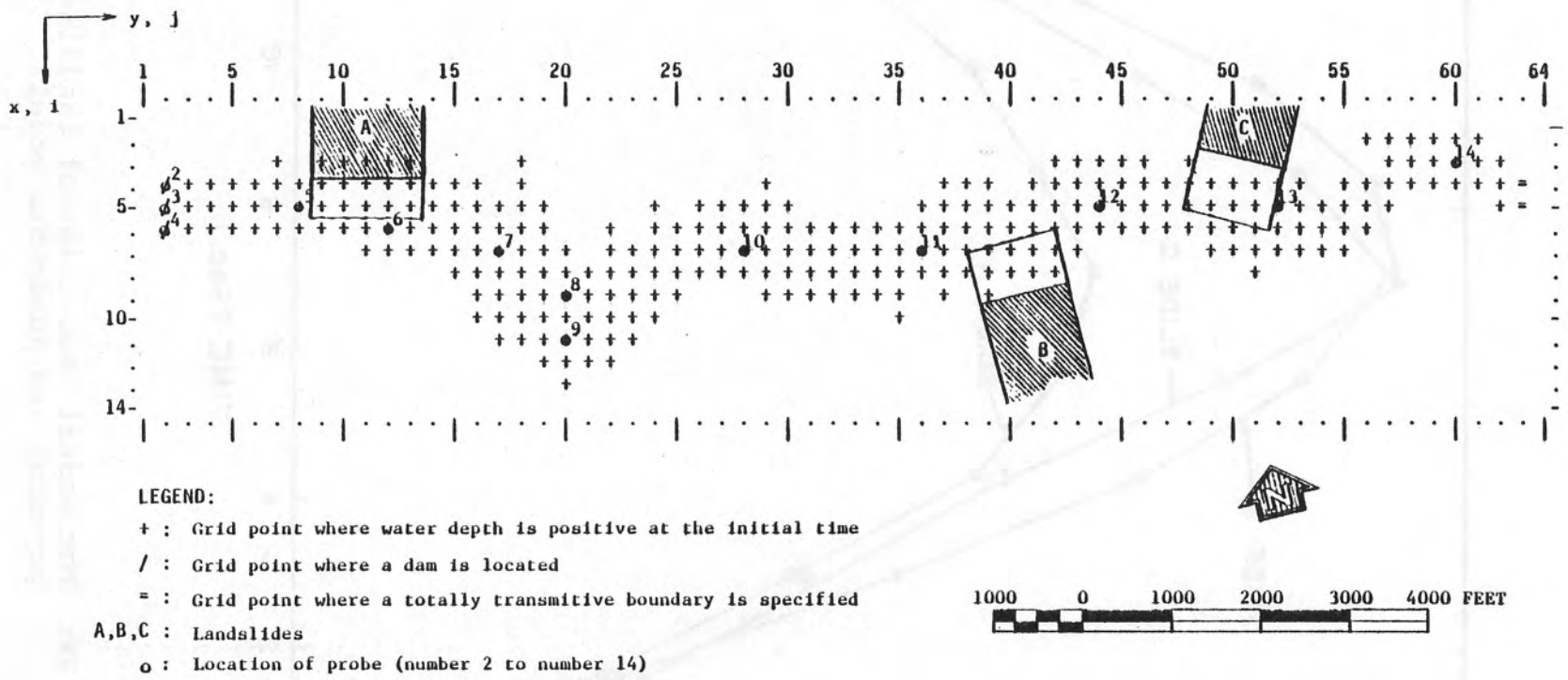
14 (Figure A7). Probes 2 to 4, from north to south, are three probes immediately upstream of Morrow Point Dam. Probe 3 is located near the center of the dam.

A3.1.2 Simulation

The numerical model developed in this study was used to simulate three laboratory runs: SLA91, SLB51, and SLC61, which are typical runs for slides "A," "B," and "C," respectively. A 14 by 64 grid was superposed over the region (Figure A7). The grid size was 250 ft by 250 ft. The orientation of the grid was chosen such that one of the two axes (x-axis in Figure A7) was parallel to the dam. An open boundary was set at the east end of the modelled reservoir. Water surface fluctuations at all 13 probe locations (number 2 to number 14 in Figure A7) were computed. Because the grid spacings are 250 ft by 250 ft, the location of a probe could be as much as 250x0.707=177 ft off from the true location in the laboratory. The computational time step is 0.6878 sec, which is half of the time step used in the laboratory to record landslide motion and water elevations. The coefficient of energy dissipation was set equal to 0.7 for SLA91 and 0.6 for both SLB51 and SLC61.

The "forced" landslide submodel was used to simulate the laboratory landslide motions. The sliding directions with respect to the x-axis were 0°, 195°, and -12°, for slides A, B, and C, respectively (Figure A7). A fine grid system based on 50 by 50 subdivisions per computational grid was used to simulate the propagation of the slides. The input time histories of slide velocity were obtained from the laboratory observations (Figure A8). About 18 time steps were required to describe the slide motions.

Time histories of the resultant water surface fluctuation at selected probe locations are depicted in Figures A9 to A16. Each of these figures consists of two plots with the upper



LEGEND:

+ : Grid point where water depth is positive at the initial time

/ : Grid point where a dam is located

= : Grid point where a totally transmitive boundary is specified

A,B,C : Landslides

o : Location of probe (number 2 to number 14)

FIGURE A7: Grid network for Morrow Point Reservoir numerical model.

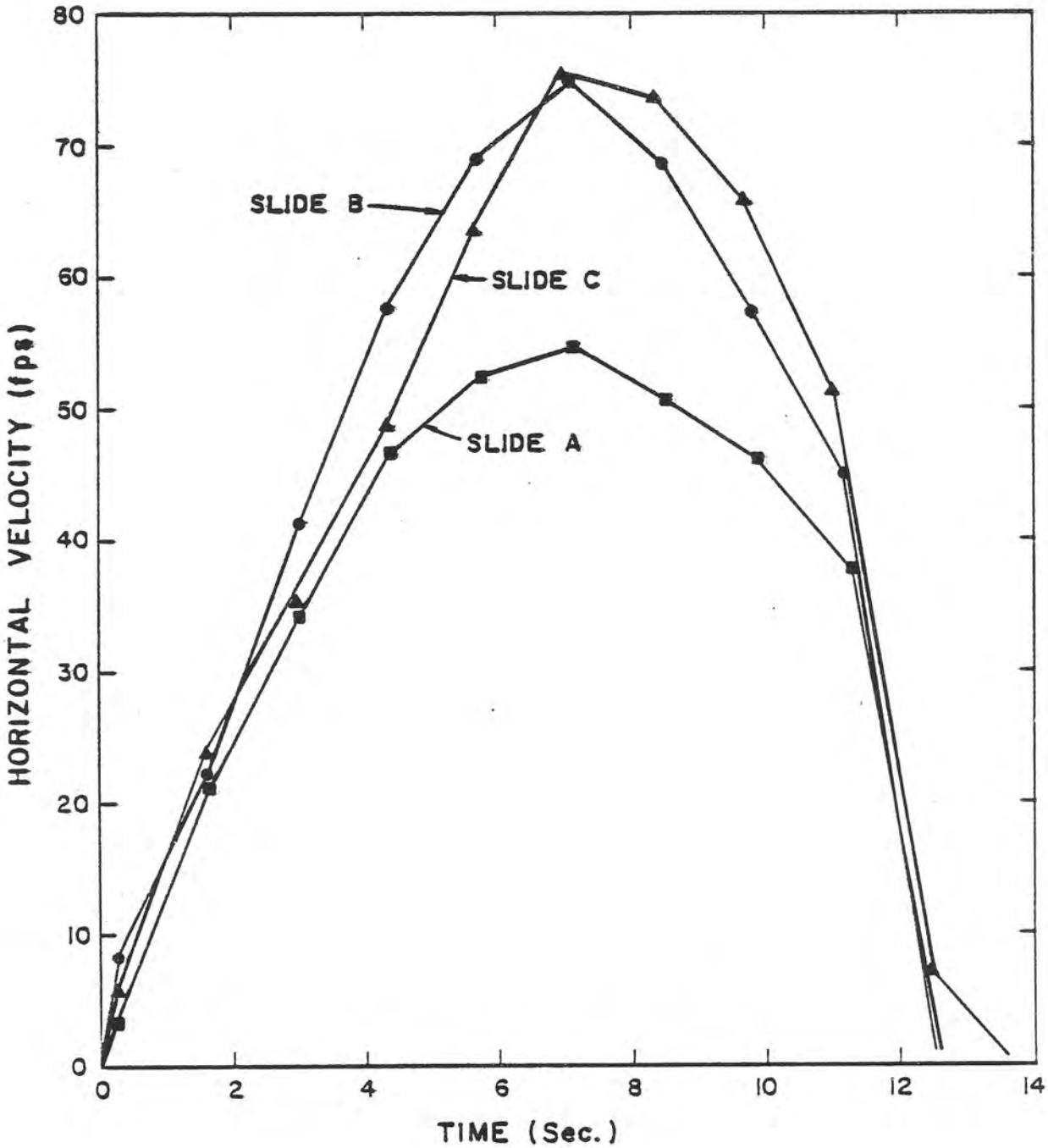
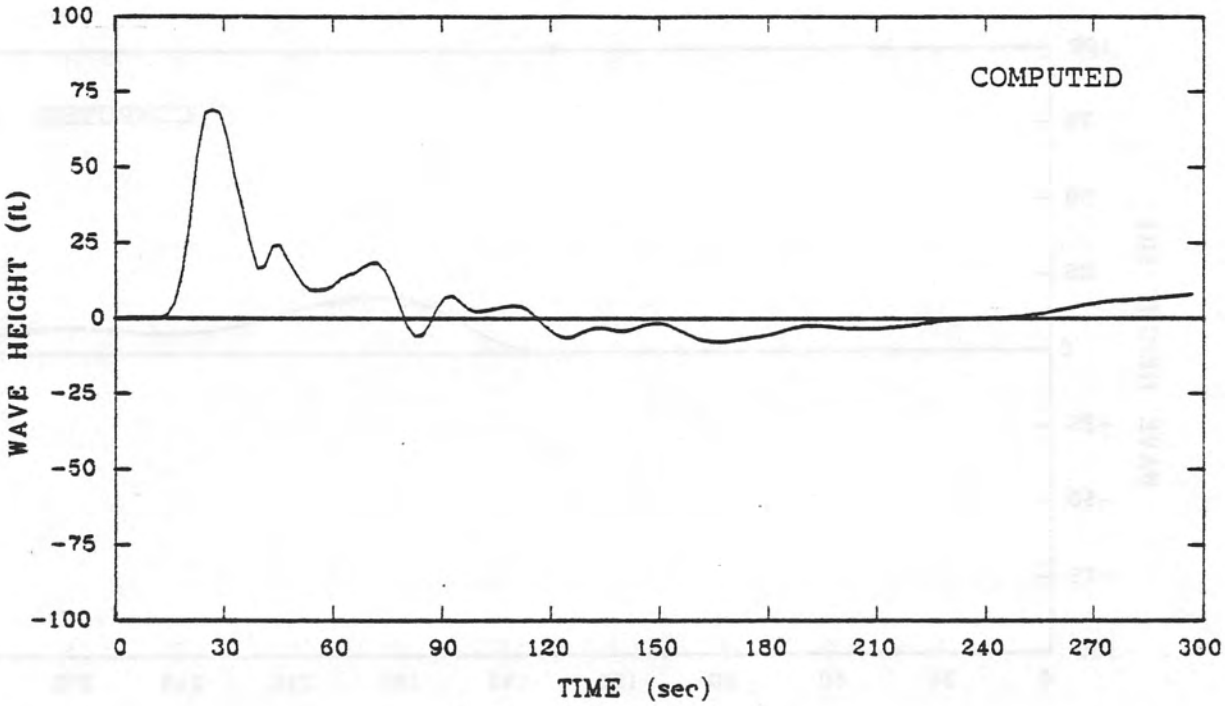


FIGURE A8: Horizontal velocities of landslides in both physical and numerical models.



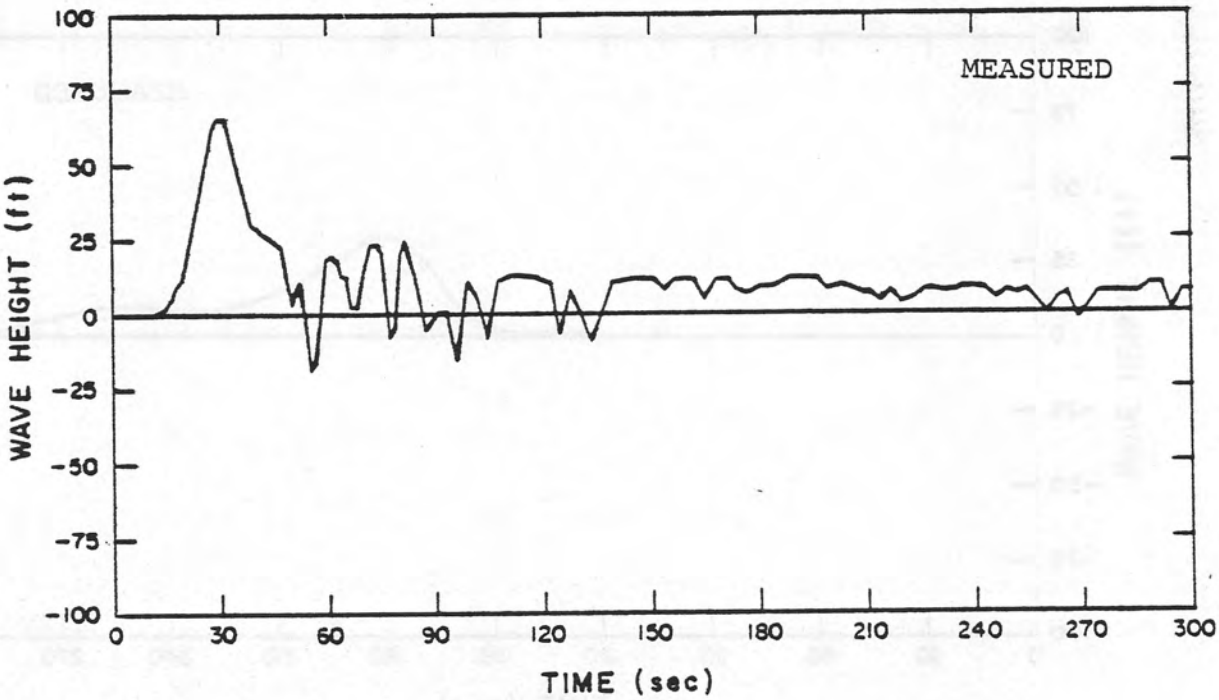
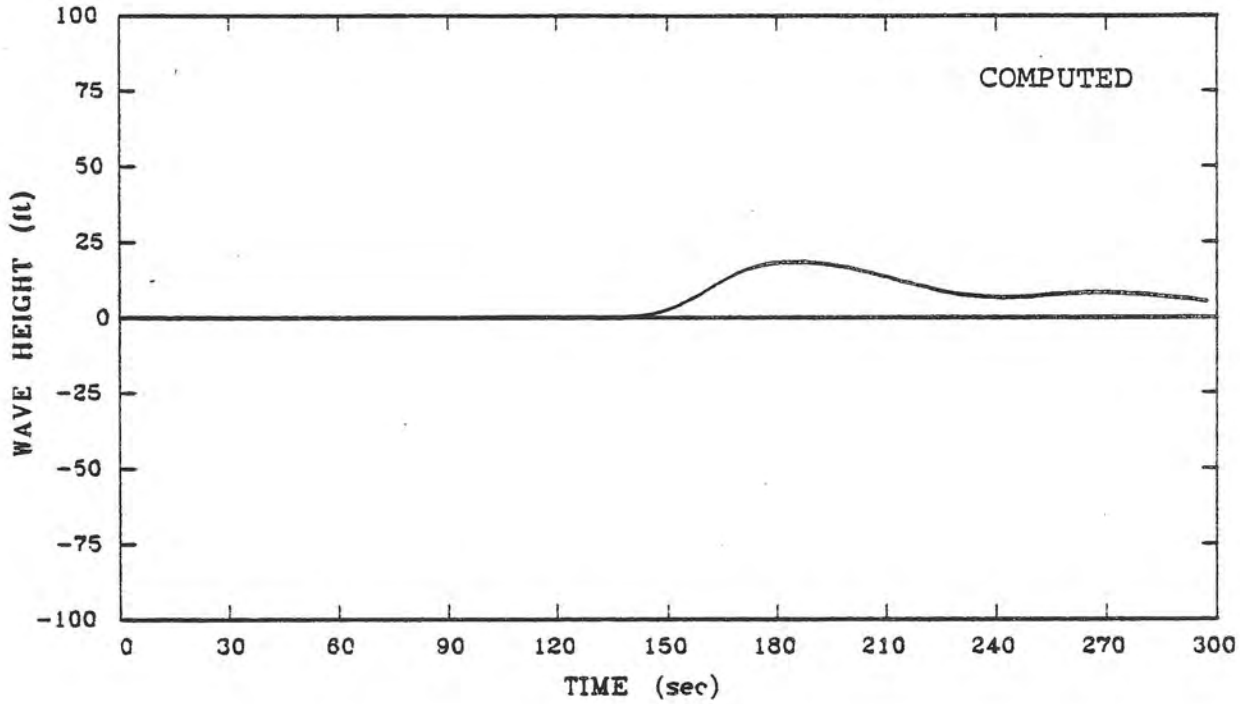


FIGURE A9: Surface fluctuation at Probe 3, due to landslide A at Morrow Point Reservoir.



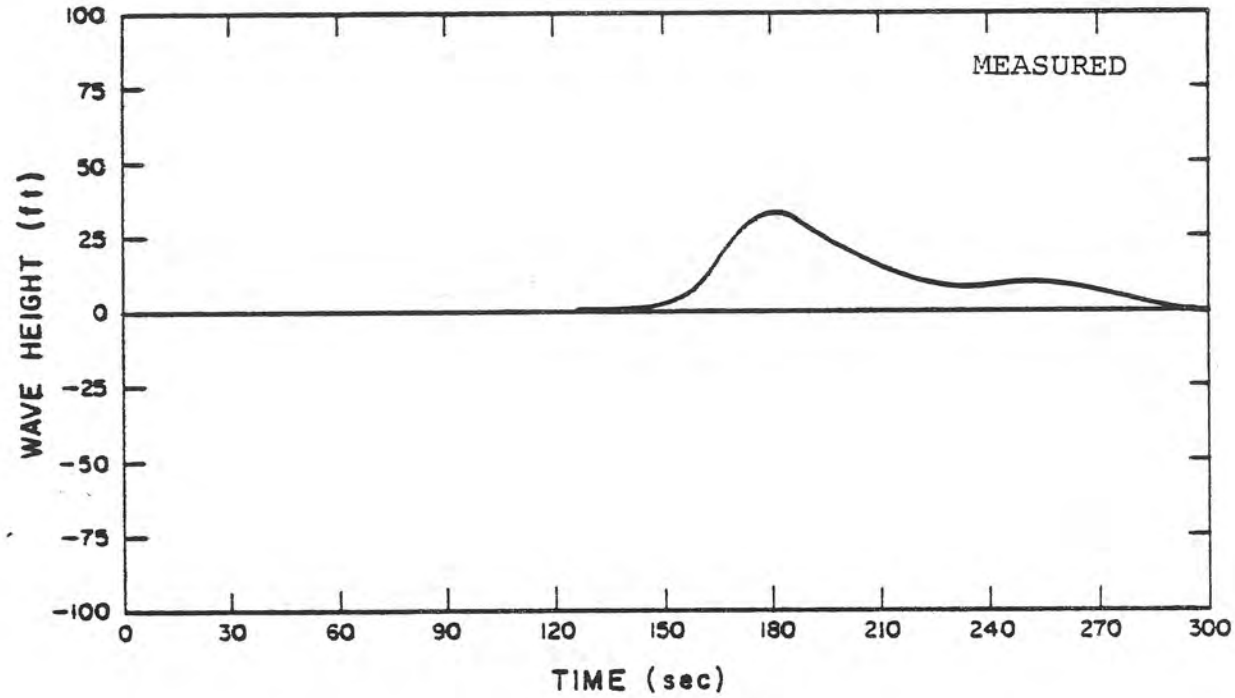
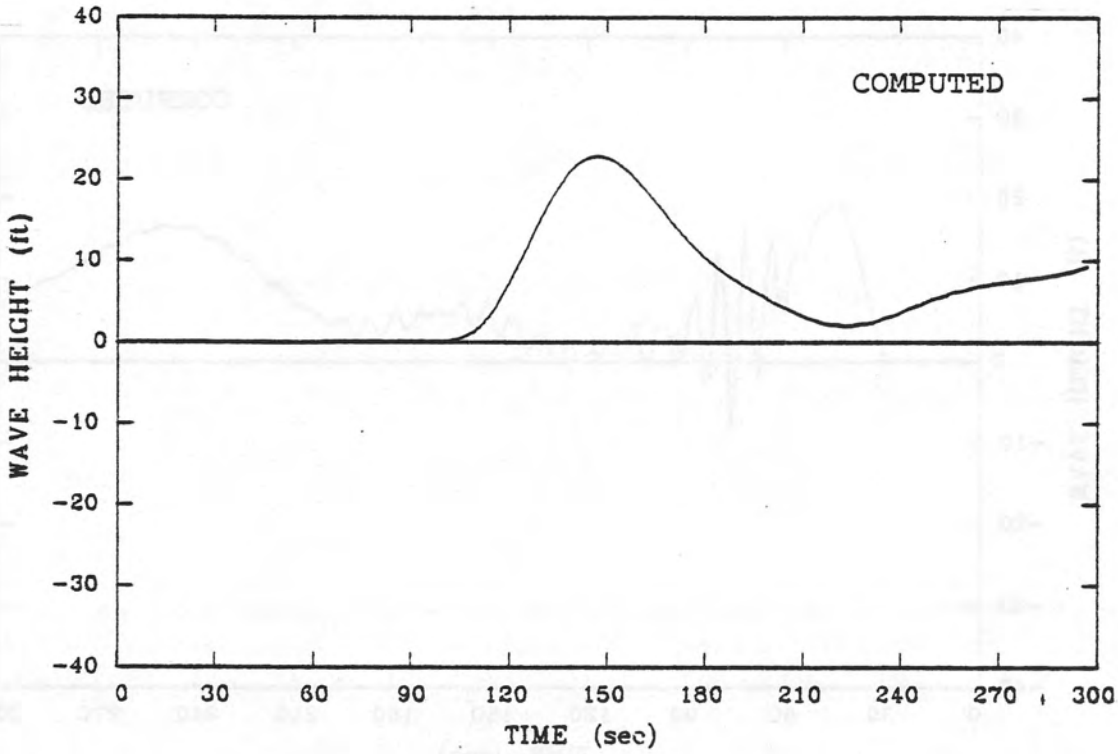


FIGURE A10: Surface fluctuation at Probe 14, due to landslide A at Morrow Point Reservoir.



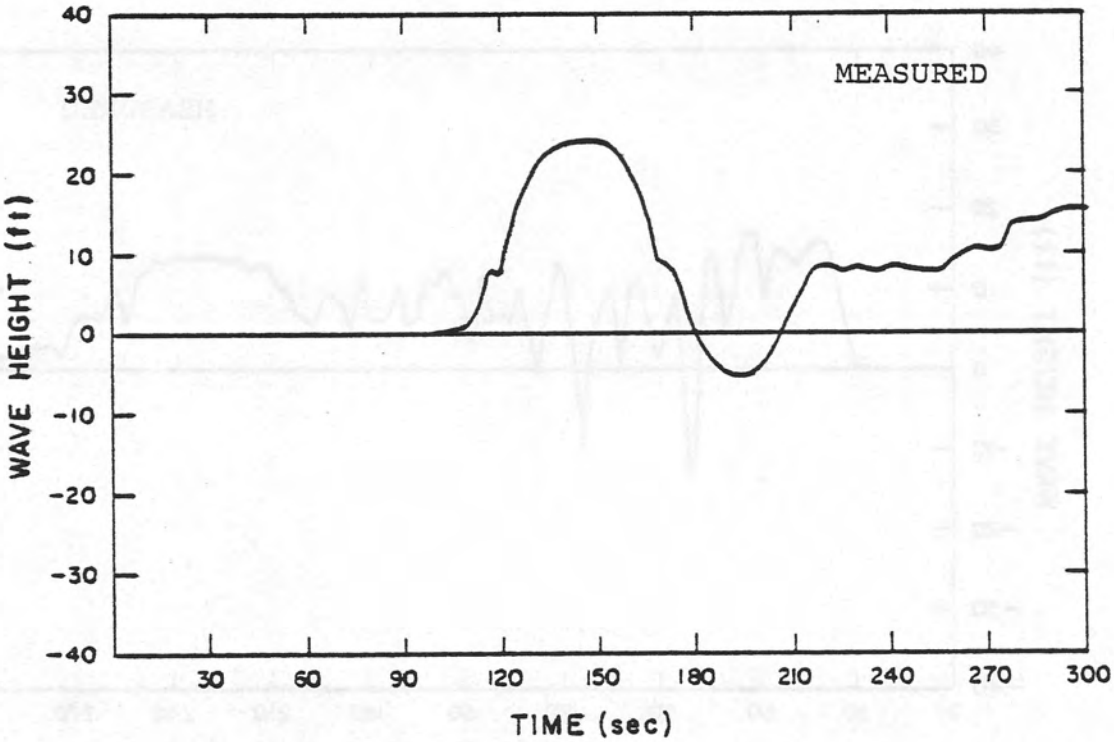


FIGURE A11: Surface fluctuation at Probe 3, due to landslide B at Morrow Point Reservoir.

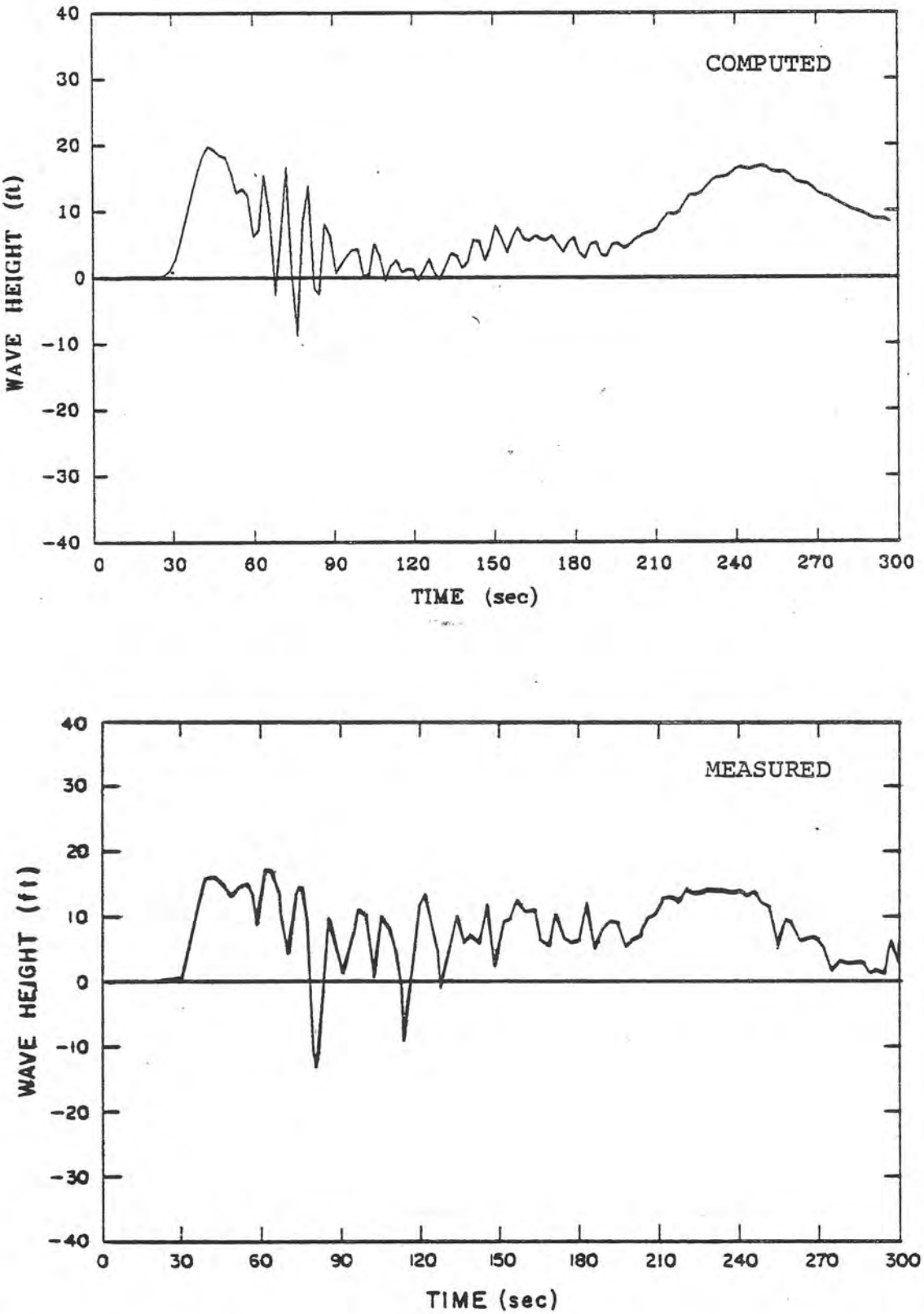
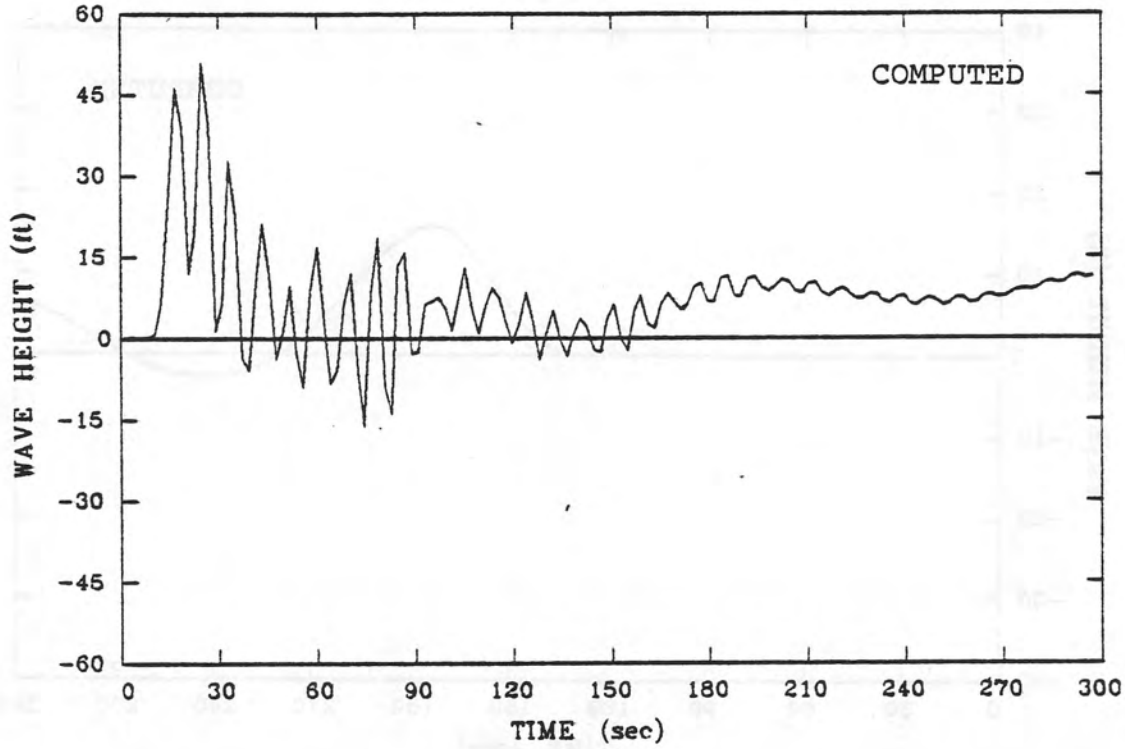


FIGURE A12: Surface fluctuations at Probe 10, due to landslide B at Morrow Point Reservoir.



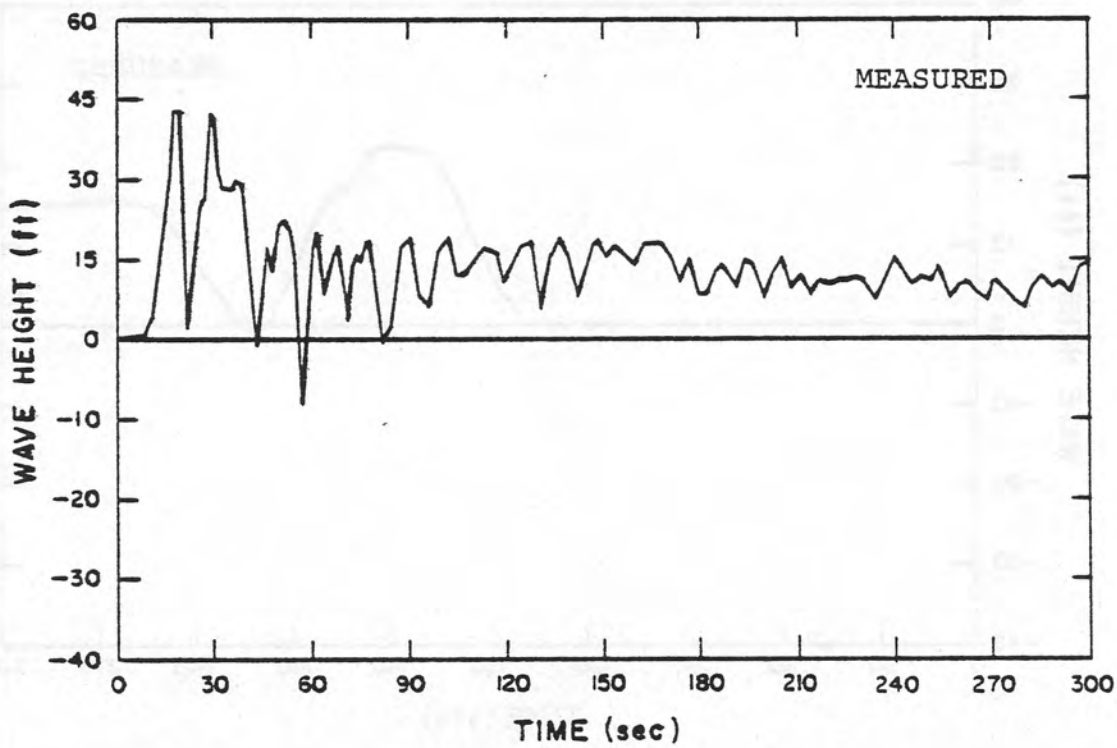
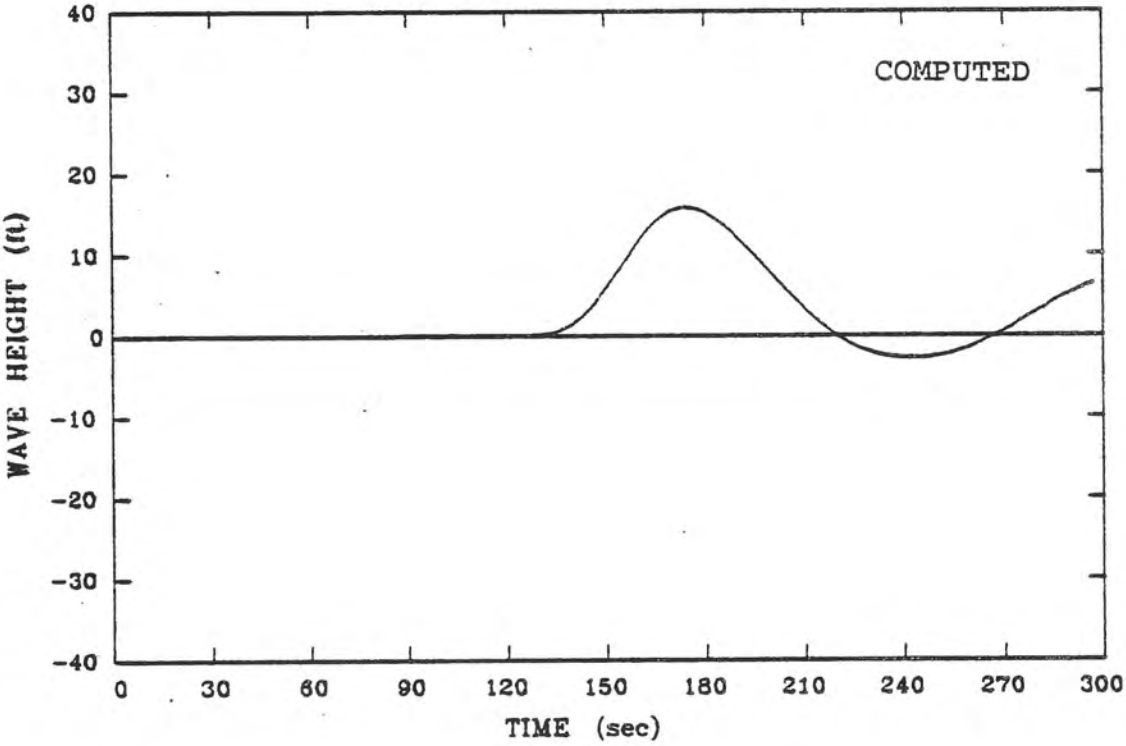


FIGURE A13: Surface fluctuations at Probe 12, due to landslide B at Morrow Point Reservoir.



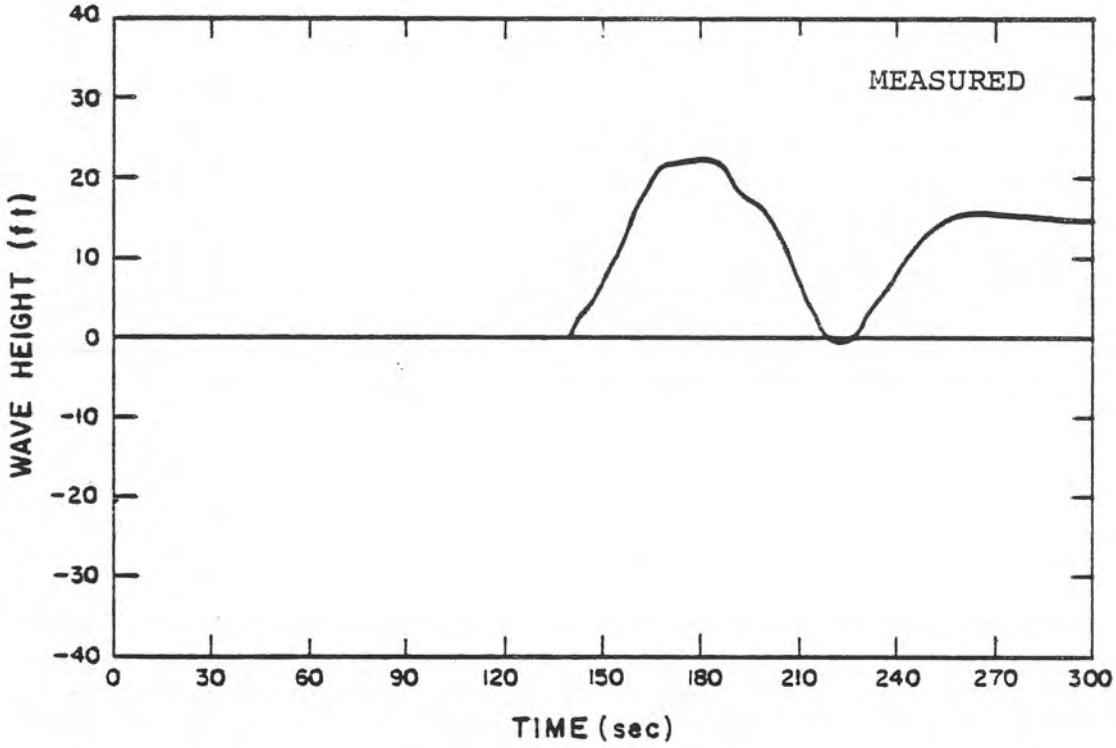
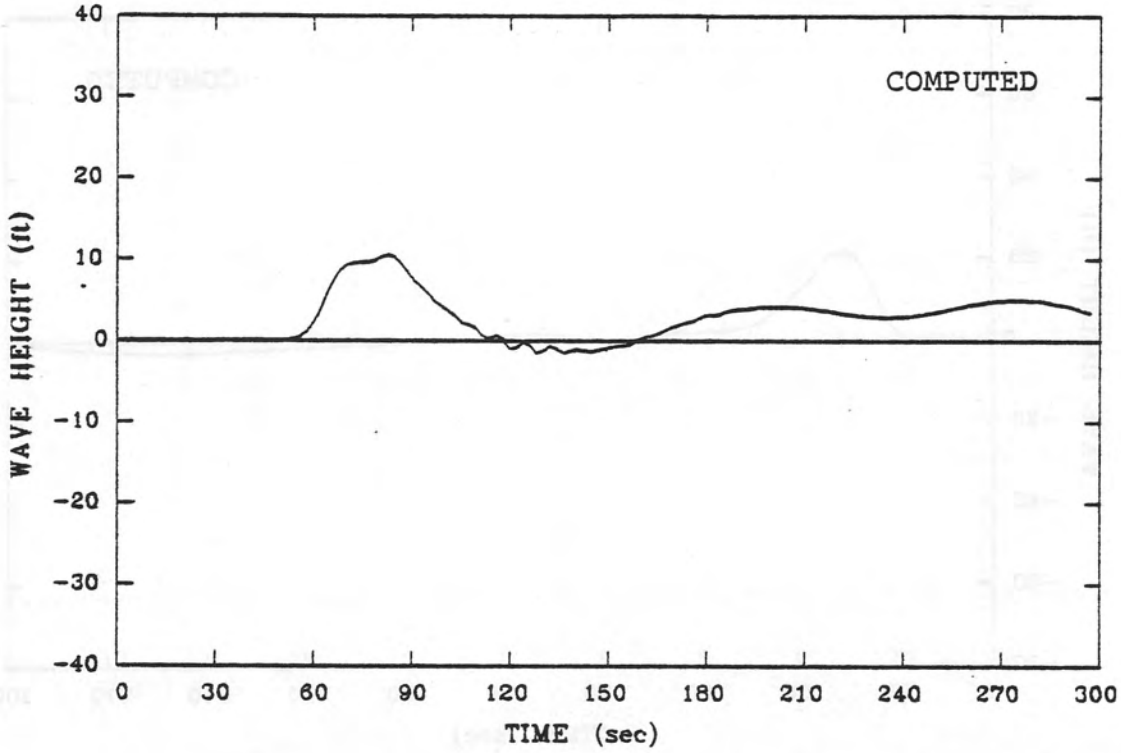


FIGURE A14: Surface fluctuations at Probe 3, due to landslide C at Morrow Point Reservoir.



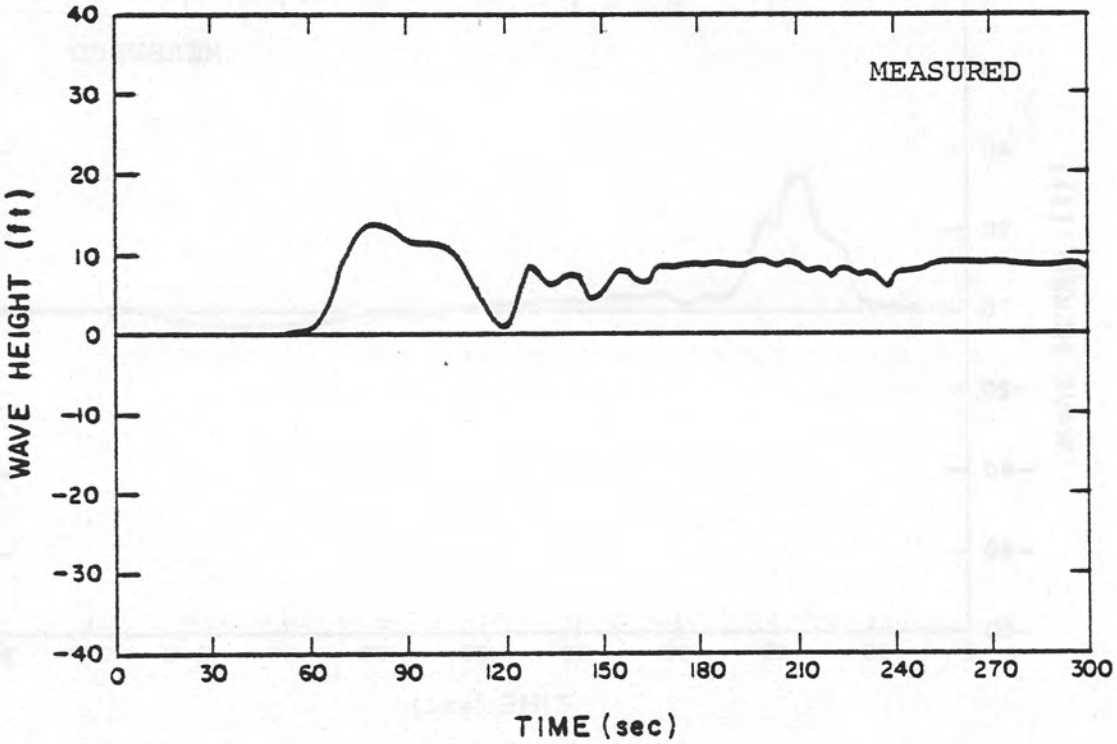


FIGURE A15: Surface fluctuations at Probe 10, due to landslide C at Morrow Point Reservoir.

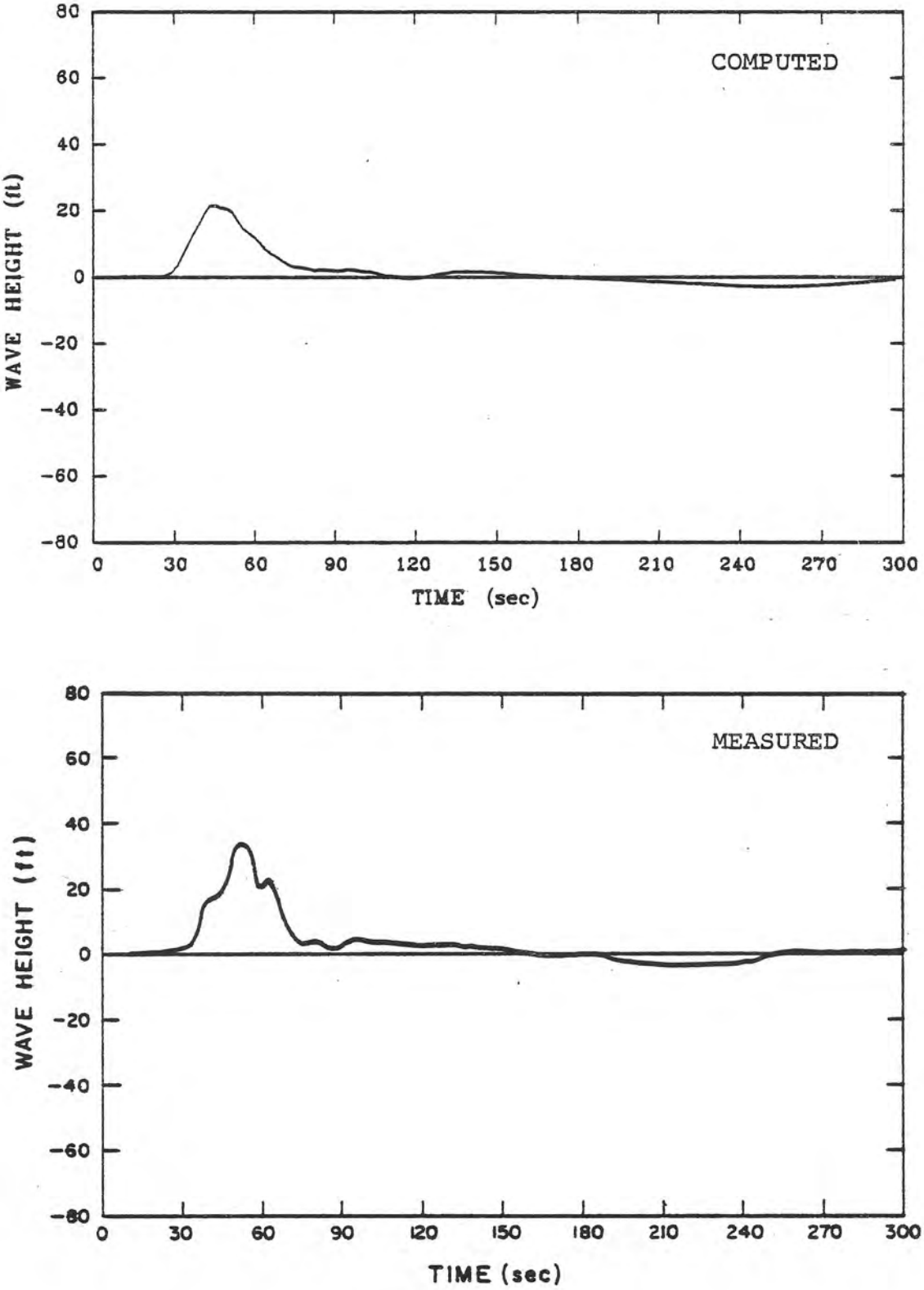


FIGURE A16: Surface fluctuations at Probe 14, due to landslide C at Morrow Point Reservoir.

one showing the numerical result and the lower one showing the laboratory result.

Figure A9 illustrates the results of slide A at Probe 3 which is located immediately upstream of the center part of Morrow Point Dam (Figure A7). The phase and peak of the first computed wave are very close to those measured in the laboratory. Figure A10 shows the results at Probe 14 which is located about three miles upstream from slide A (Figure A7). After turning around a river bend and traveling for about two miles, the original huge wave becomes a long undulation. The computed wave height is a little lower than the measured one. The fluctuations of water surface after the first wave peak has passed depend on how precisely the boundary and bathymetry are represented in the model. Normally, it is meaningless to check too many detail after the first or second peak. However, the general patterns in Figures A9 to A16 are all extremely encouraging.

When the full nonlinear model was used to simulate the waves generated by slide B, a nonlinear numerical instability occurred, probably due to the high slide velocity (Figure A8), and certain combinations of geometry and bathymetry. Therefore, the nonlinear terms in the governing equations were surpressed in this run (a convenient user option). Figure A11 depicts the surface fluctuation at Probe 3, which is located at the center of the dam (Figure A7). Again, the wave peak, phase, and general pattern all show good agreement despite the approximation. Figure 12 shows the results at Probe 10 which is located at the middle section of the reservoir (Figure A7). Short period fluctuations indicate the effects of wave reflection from boundaries and refraction due to the uneven bottom. Figure A13 shows the results for Probe 12 near slide B. A little higher peak from the numerical model could be due to the error of the probe location, since the wave changes extremely rapidly near the landslide area.

For slide C, Figure A14 shows the computed and measured surface fluctuation at Probe 3 near the center of the dam. The phase and shape of the first wave agree well. The computed wave has a slightly smaller amplitude. At the middle section of the reservoir, where Probe 10 is located, the results are illustrated in Figure A15. There are a few seconds of phase error. At the eastern end of the reservoir, the results from Probe 14 are depicted in Figure A16. The computed phase and general pattern agree with the measured ones. The computed wave amplitude is slightly smaller than the measured amplitude. The kink in the first measured wave probably is caused by partial reflection from the damping boundary at the eastern end of the physical model.

When initial values of free-board at Morrow Point Dam for slides A, B, and C of 5.75 ft, 4.00 ft, and 4.25 ft, respectively, the computed discharges over the dam are depicted in Figure A17. The peak discharge due to slide A reaches 1.7 million cfs, while those due to slides B and C are only 0.2 million cfs and 0.1 million cfs, respectively. The main reason for smaller discharge is wave decay and dispersion over the longer paths. A minor reason is that slides B and C are both of four-fifths of the size of slide A. As a final comparison, the total volume of overtopping for Slide A (SLA91) was computed to be 22.6 million cubic feet with a discharge coefficient of 4.0. The measured volume was 21.9 million cubic feet.

In conclusion, the numerical model in this study is capable of reproducing the landslide-generated water waves tested in the laboratory. Using landslides at all three locations, the comparison between computed results and measured results is very good.

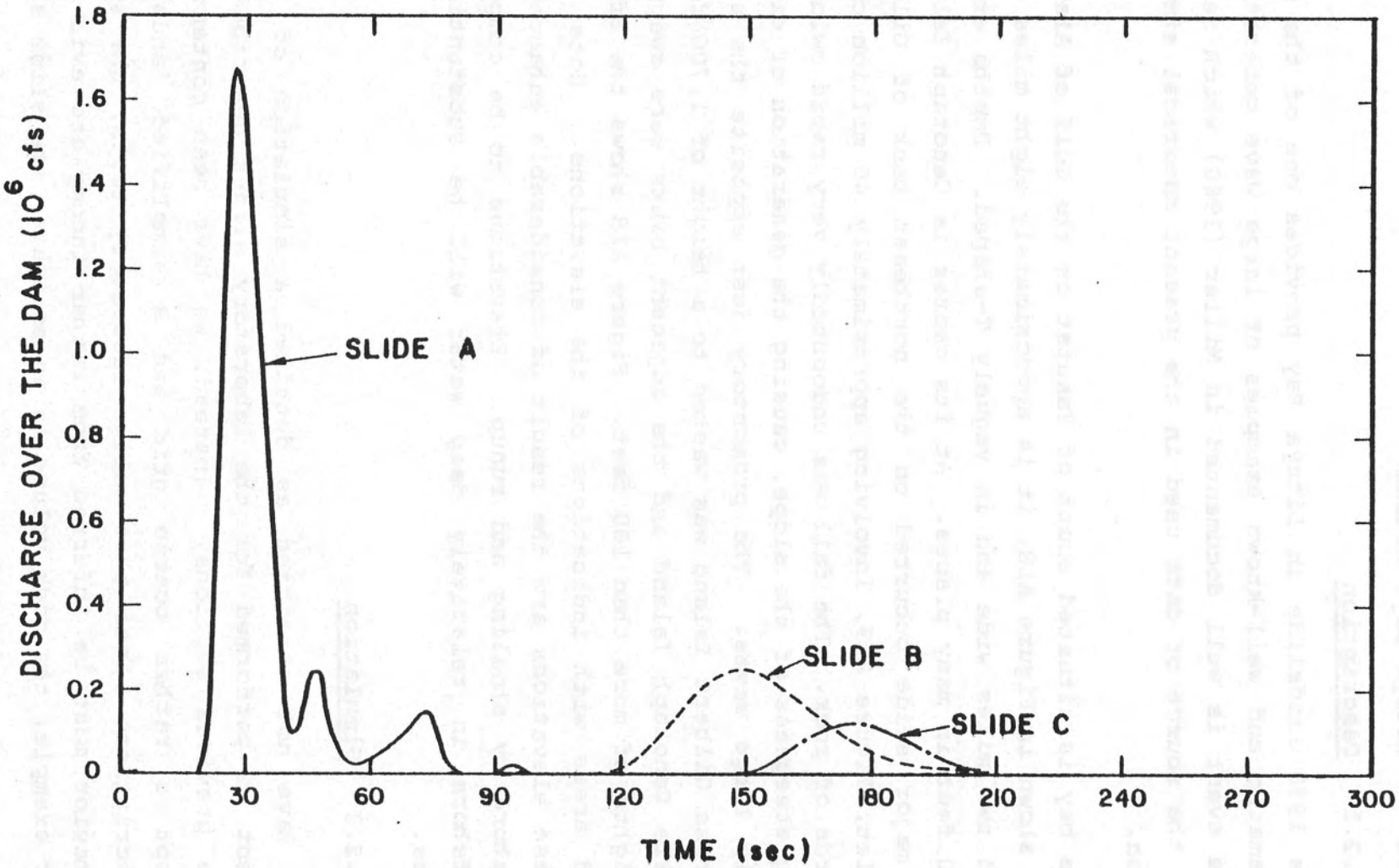


FIGURE A17: Computed discharge hydrograph at Morrow Point Dam.

A3.2 Lituya Bay, Alaska

A3.2.1 Description

The 1958 landslide in Lituya Bay provides one of the most dramatic and well-known examples of large wave generation. The event is well documented in Miller (1960) which served as the source of data used in the present numerical simulation.

The bay is situated south of Yakutat on the Gulf of Alaska. As shown in Figure A18, it is approximately eight miles long and two miles wide and is vaguely T-shaped. Depths exceed 600 feet in many places. At its center is Cenotaph Island. A major slide occurred on the northeast bank of Gilbert Inlet, Figure A19, involving approximately 40 million cubic yards of rock. The fall was undoubtedly very rapid owing to the steepness of the slope, causing the generation of extremely large waves. The promontory just opposite the slide across Gilbert Island was washed to a height of 1,700 feet, while Cenotaph Island and the adjacent banks were swept to heights of more than 100 feet. Figure A18 shows the affected areas with indications of the elevations. Note that these elevations are the result of considerable enhancement onshore by shoaling and runup. Elevations to be computed offshore in relatively deep water will be substantially less.

A3.2.2 Simulation

We have not attempted as detailed a simulation of this event as performed for the laboratory slides described in the previous sections. Instead, we have been content to adopt a rather coarse grid and a simplified landslide description. Note that unlike laboratory tests, the slide behavior must be inferred from rather incomplete evidence. For example, the rock volume is an estimate, the slide shape

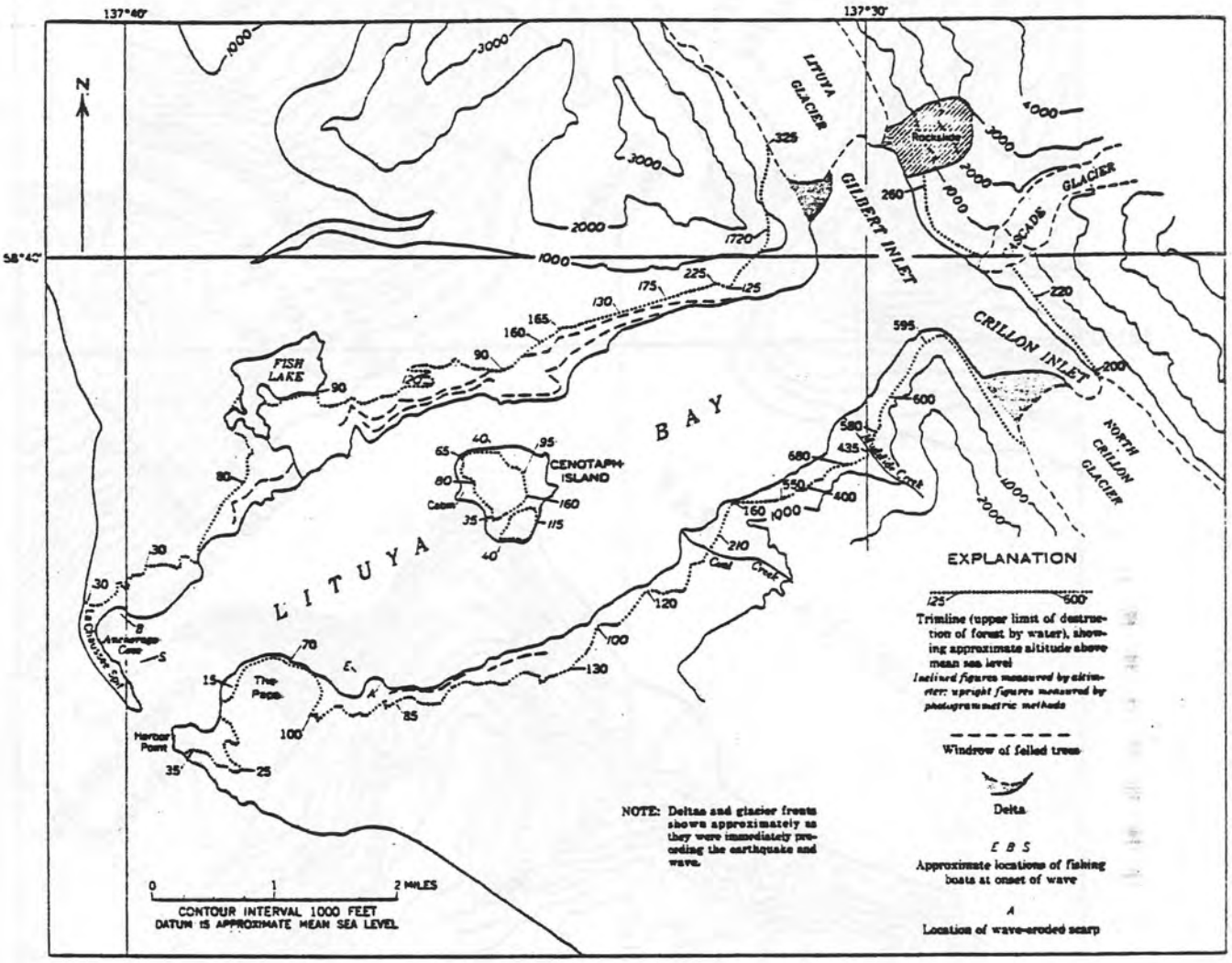
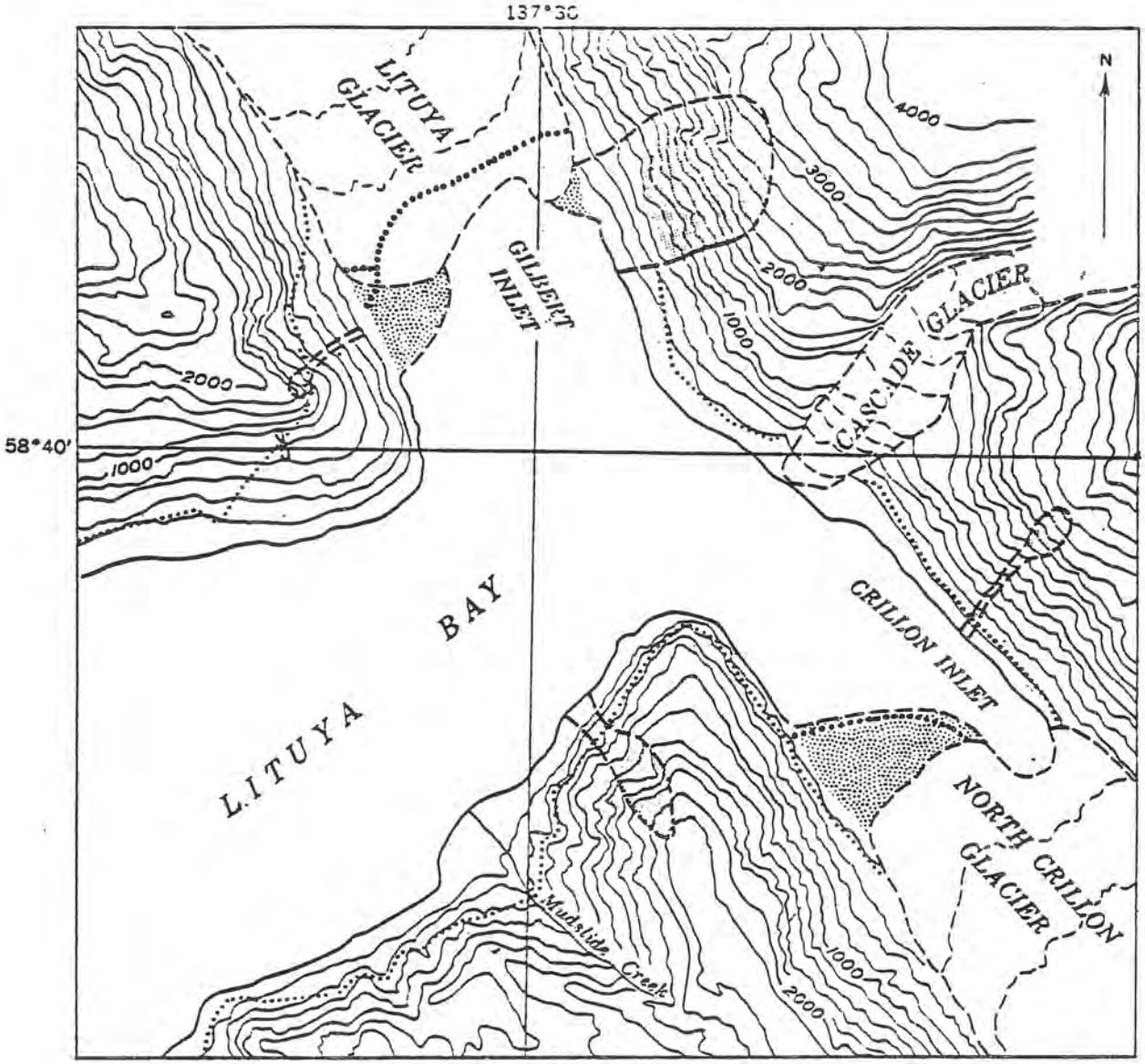


FIGURE A18: Lituya Bay. Dotted line and associated small numbers show extent and elevation of region affected by waves. (From Miller, 1960).



Trimline (upper limit of destruction of forest by water)

0 ½ 1 MILE

FIGURE A19: Upper reach of Lituya Bay showing approximate location of the Gilbert Inlet slide (region east of Gilbert Inlet enclosed by dashed line). (Adapted from Miller, 1960).

is not known, and it is unknown to what degree ice may have participated in the displacement.

The simulation described here is correspondingly simple. The space step size was taken to be a rather coarse 1,000 feet in the interests of economy. The slide was modeled using the natural landslide option in two ways in order to test the sensitivity of the results to slide shape. Referring to Figure A19 in which the slide area is delineated, it is seen that the slide width is about 2,500 feet and the region extends up to the 3,000 foot elevation contour over a horizontal distance of about 3,800 feet. Consequently, our model slide was taken to be a 2,500 x 3,800 foot rectangle in plan view. Two assumptions were made regarding the slide thickness. In the first, we assumed a simple wedge shape with a blunt leading edge. In the second, we assumed a more streamlined shape tapered at the leading and trailing edges and thickest near the middle.

The adopted shape tables for these two slides were:

WEDGE		TAPERED	
XSL	HSL	XSL	HSL
0	227.4	0	0
3800	0	1016	136.5
		1988	182.0
		2916	136.5
		3799	0

These shapes each represent a total volume of 40 x 10^6 cubic yards in accordance with estimates. The variables XSL and HSL, used in Program LSWAVE, are horizontal distance back from the leading edge of the slide, and vertical thickness of the slide mass, respectively.

The landslide was taken to occur directly downslope with a friction angle of 10°. The coefficient of energy dissipation was set equal to zero representing a conservative ease

of fully reflective walls, and the specific gravity of the mass was taken to be 2.6 corresponding to rock. Other parameters such as drag coefficients were set to default values discussed in the users' manual (Part B). A complete set of input data has been supplied to the Bureau of Reclamation as a file in a magnetic tape accompanying this report. The velocity time histories of the two slides were found to be:

WEDGE		TAPERED	
TIME TSL	VELOCITY VSL	TSL	VSL
0	0	0	0
14.1	134.5	13.3	143.2
20.4	170.5	18.9	191.1
25.8	176.4	23.6	211.9
31.5	157.2	28.2	208.0
38.4	118.0	33.0	184.8
50.3	42.6	38.7	146.7

Units in all tables are seconds and feet. Note that the indicated velocities are the horizontal component of the velocity of the leading edge.

The two slides behaved qualitatively as expected. That is, the more streamlined tapered shape achieved greater speeds than did the blunt wedge. Both slides show an initial acceleration followed by a deceleration and eventual halt. Adjustment of slopes, friction, shape, and so forth would permit the modeler to achieve a variety of displacement histories and a corresponding range of resulting wave conditions.

The results are summarized in Table A1 which shows maximum computed wave elevations at a number of points within the bay. The locations of the points are indicated in Figure A20.

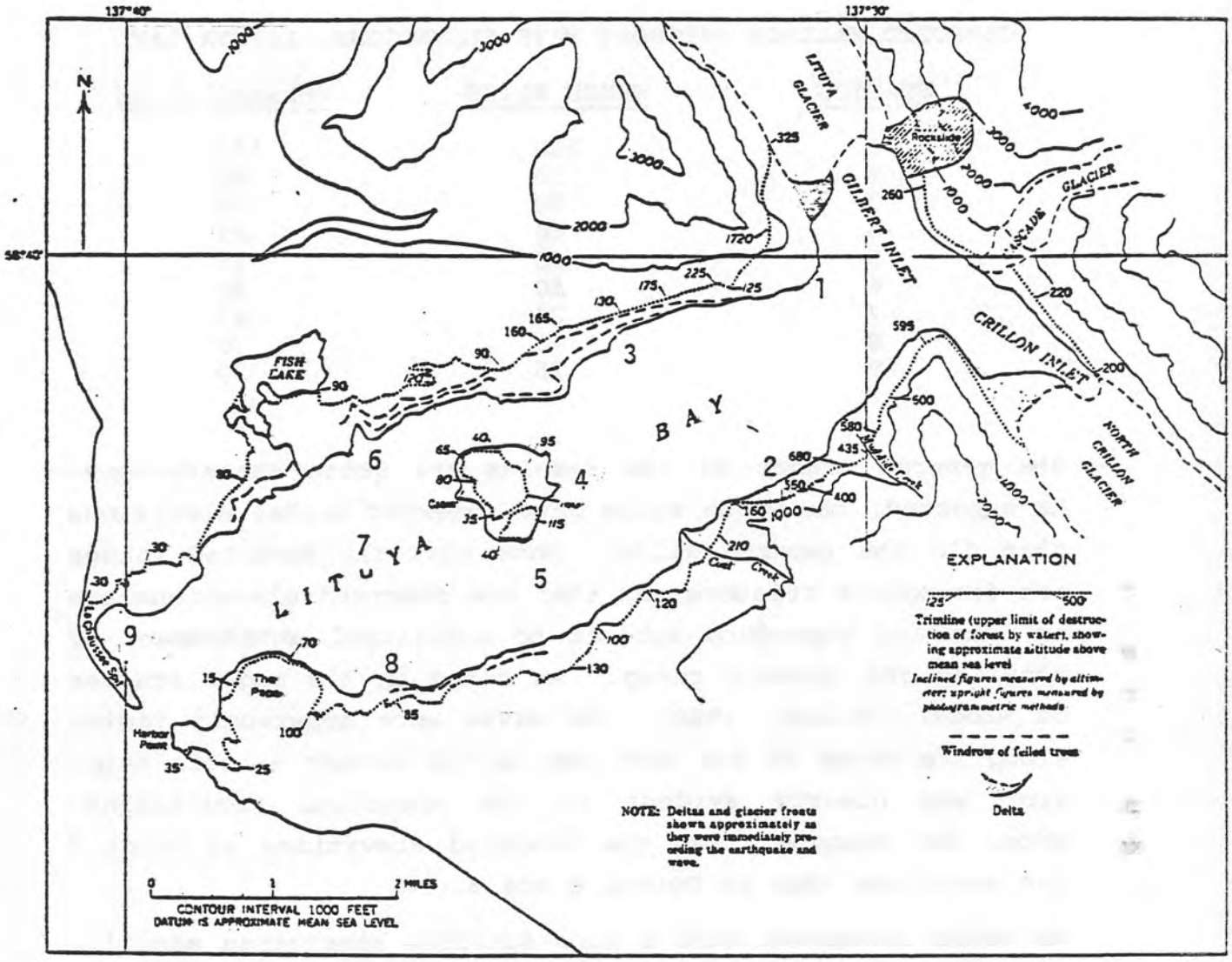


FIGURE A20: Map showing locations of nine stations for which computed wave elevations are showing in the accompanying table.

TABLE A1

COMPUTED MAXIMUM OFFSHORE WAVE ELEVATIONS, LITUYA BAY

STATION	WEDGE SLIDE	TAPERED SLIDE
1	200	120
2	120	80
3	80	70
4	60	45
5	30	25
6	50	40
7	20	12
8	40	30
9	25	20

The general trends of the results are quite satisfactory. As expected, the wedge slide gave somewhat higher elevations than did the tapered slide. Note that all computed values are for points offshore and that the observed elevations are on land and therefore subject to additional enhancement by shoaling and dynamic runup. As noted in the model studies of Wiegel (Miller, 1960), the waves were apparently higher along the edges of the bay than in the center -- this behavior was clearly evident in the numerical simulations. Note, for example, that the computed elevations at Point 7 are much less than at Points 6 and 8.

We would recommend that a more detailed simulation adopting a finer step size and, perhaps, more detailed treatment of the slide, should be performed. A grid with a step size of 250 feet would be ideal since the wave system would be resolved much more accurately. The present simulatin is at the limit of large space step.

A3.3 Mica Reservoir--Example of an Earth Dam

A3.3.1 Description

Mica Reservoir is situated on the upper Columbia River in southeast British Columbia. The dam is an earth structure

approximately 650 feet high with a base thickness of about 3000 feet and a crest length of 2700 feet. The large reservoir is bounded by a number of potential slide areas causing concern that--should a major slide occur--the associated waves could overtop and erode the dam, leading to rapid failure.

In order to investigate the extent of the potential hazard, Western Canada Hydraulic Laboratories conductred an extensive series of physical model tests at a scale of 1:300. The study report (Western Canada Hydraulic Laboratories, 1970) served as the source of all data used in the present simulations.

Several potential slide areas were identified for testing including Dutchman's Ridge, Little Chief Creek, Molson Ridge, and Franchere Creek. The latter two sites lie across the major reservoir at a considerable distance from the dam. The site at Little Chief Ridge lies relatively near the dam and was the subject of relatively detailed tests in the model study. Therefore, we have chosen the Little Chief Ridge slides for numerical simulation here.

A3.3.2 Simulation

At the outset, it should be noted that there is some ambiguity in the WCHL report regarding quantitative features of the model and the slides. Several figures must be schematic since slide locations, shapes, shoreline, and so forth do not always agree between two figures. It has been necessary to infer much of the information used in the numerical simulations. Nevertheless, the results are in sufficient agreement with the laboratory tests to indicate the general validity of the model.

Figure A21 shows the region modeled in the numerical study. Also shown shaded is the approximate location of the Little

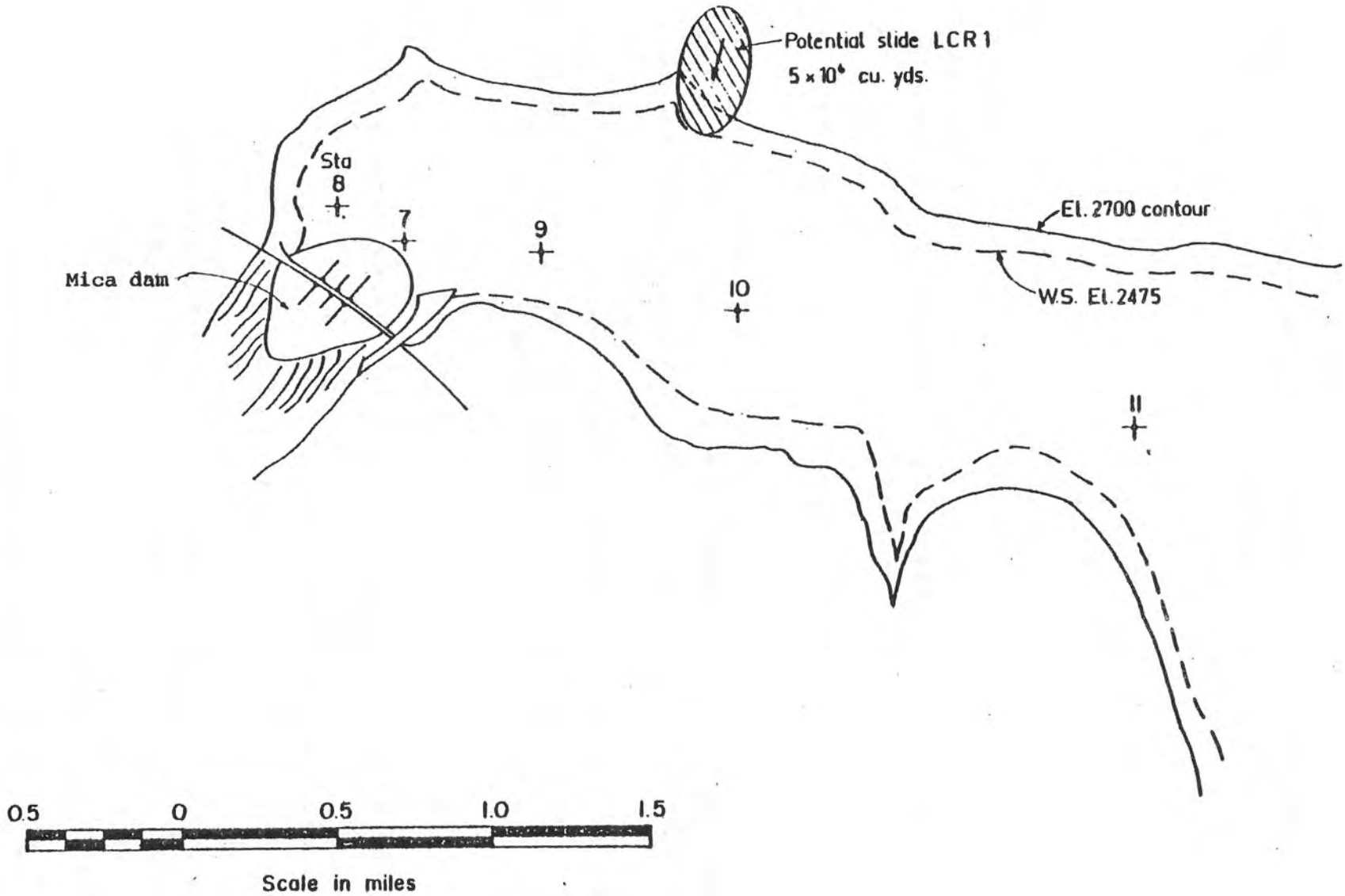


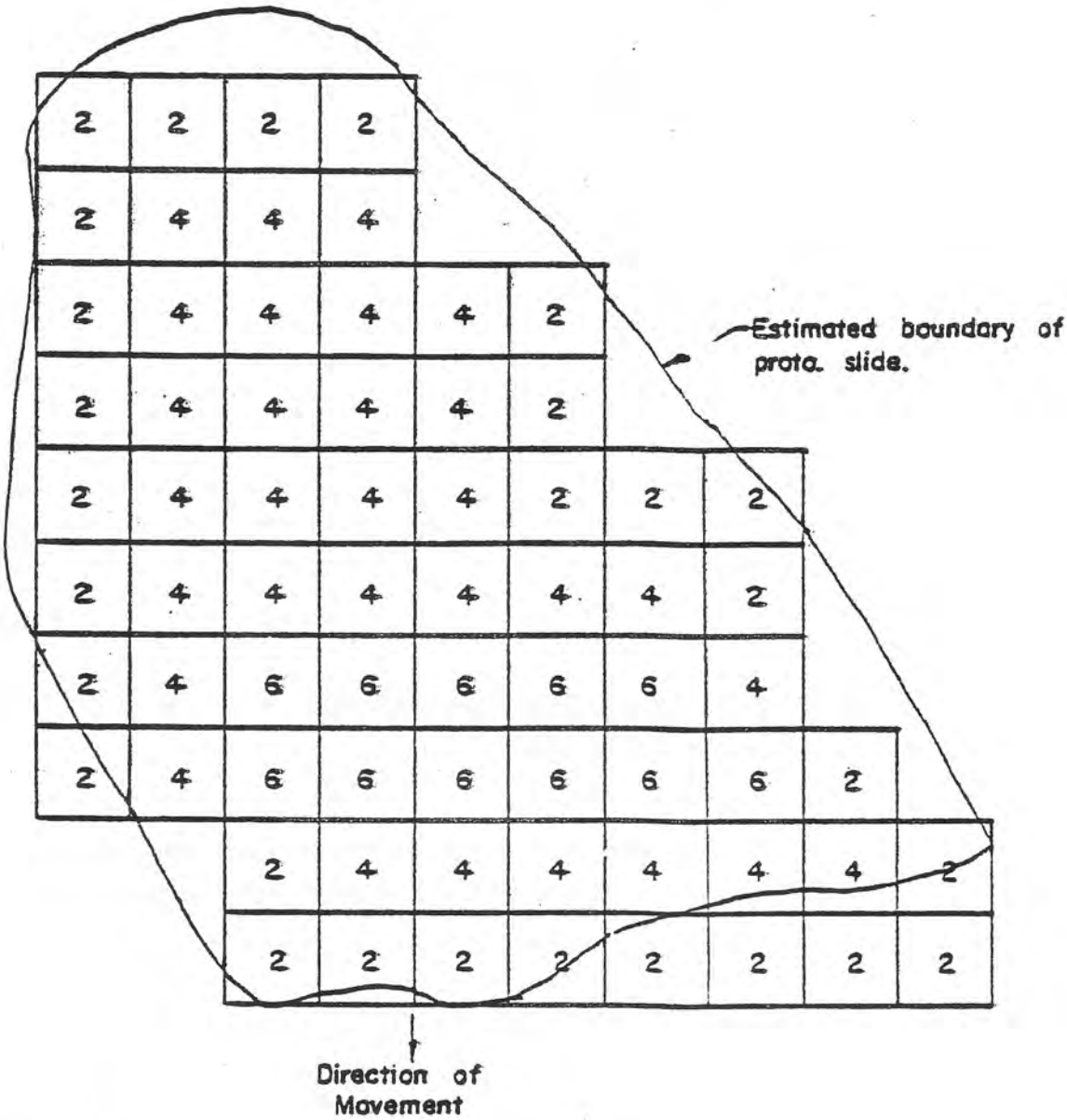
FIGURE A21: Configuration of Mica Reservoir near Mica Dam showing position of Little Chief Ridge landslide and wave gage stations 3, 7, 9, and 10. (Adapted from Western Canada Hydraulic Laboratories, 1970).

Chief Ridge slide LCR-1. A finite difference grid with a space-step of 500 feet was adopted, with the dam crest lying along a coordinate direction. The energy dissipation coefficient was 0.7 and the slide bulk density was 2.0. The complete data deck describing all input has been supplied to the Bureau of Reclamation as a file on a magnetic tape accompanying this report. Note that the slope of the face of the dam was accounted for in the depth grid with the chosen space step. A larger step would cause the dam slope to "disappear" from the numerical model. In general, when modeling an earth dam, the upstream face should cover several grid points, say five or more, if the shoaling behavior of the waves over the dam is to be represented in the numerical results. More points would provide better resolution, to be weighed against increased computer costs.

Two sorts of laboratory simulations of the Little Chief Ridge slide were performed:

(a) In the initial laboratory test series, the model slide was a simple rectangular assembly of gravel bags of constant thickness. The bag-mat was 4 feet square with a thickness of 4 inches representing a slide volume of 5.33 million cubic yards. This model slide mass entered the reservoir via a steep, smooth ramp (45° incline). By varying the initial elevation of the gravel mat, slides of varying entry velocity could be produced.

(b) The revised test series adopted an entry ramp of adjustable slope (45°-75°) so that higher entry velocities could be achieved without raising the slide mass. In addition, the slide mass was extensively modified as shown in Figure A22. The figure shows a large number of small squares each representing a 6 inch square gravel bag.



NOTE

- Figures are thickness of slide measured in model inches.
- Each square measures 6" x 6" model.

FIGURE A22: Geometry of Little Chief Ridge landslide in lab Series 2. Note that the model scale was 1:300. (Adapted from Western Canada Hydraulic Laboratories, 1970).

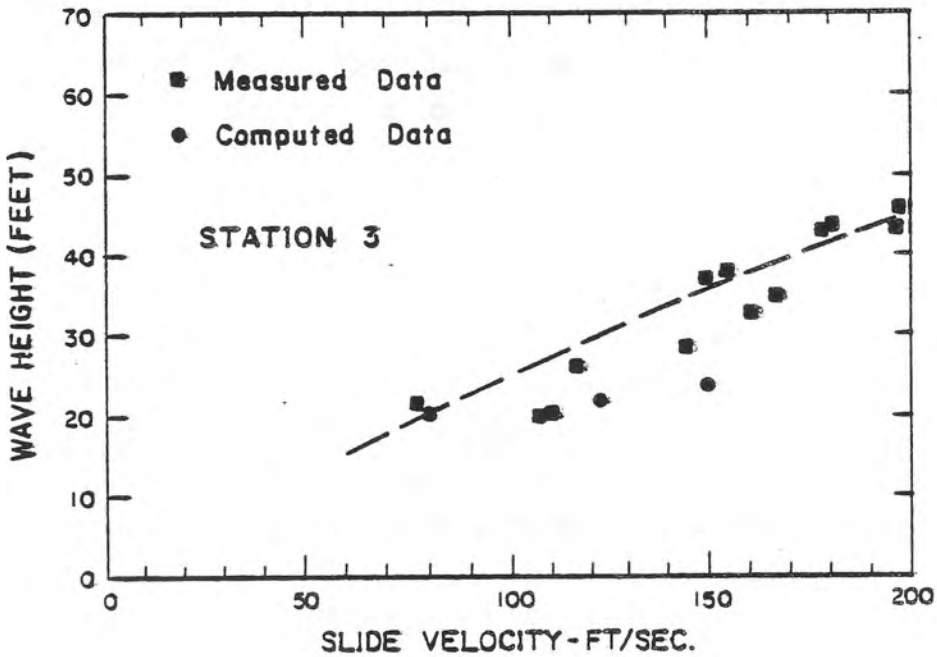
The numbers within the squares represent the individual bag thicknesses in inches. The total simulated slide volume at the 1:300 scale is 5 million cubic yards.

We shall refer to these two test series as Series 1 and 2, respectively. In addition to the just described differences in the model slides, there were significant differences in the recording and presentation of data. In fact, the data collected in Series 1 appears to have been more extensive and more useful than that taken during Series 2. During Series 1, wave gages were placed near the dam crest, near the toe of the dam, and at points between the slide and the dam. Four gage locations for Series 1 to be used in later comparisons are indicated on Figure A21.

During Series 2, only seven gages were used, including six near the dam crest and one upstream from the slide. Consequently, no data between the slide and the dam is available for comparison here.

In order to simulate Series 1, we reproduced the slide shape (a simple block) and ran several cases of the "natural" slide option corresponding to different starting elevations. For each case, the program output included a velocity time history of the slide motion. From this, it was a simple matter to determine the slide velocity at the moment the mass first entered the water (the reservoir level was taken to be 2475 ft in all tests). This permitted the determination of amplitude of the leading wave at selected gage points versus entry velocity, which was the format of data presentation used in the WCHL report. Note that the program LSWAVE shows the horizontal component of the slide velocity--this was adjusted to yield the along-slope value used by WCHL.

The numerical results are summarized in Figures A23 and A24



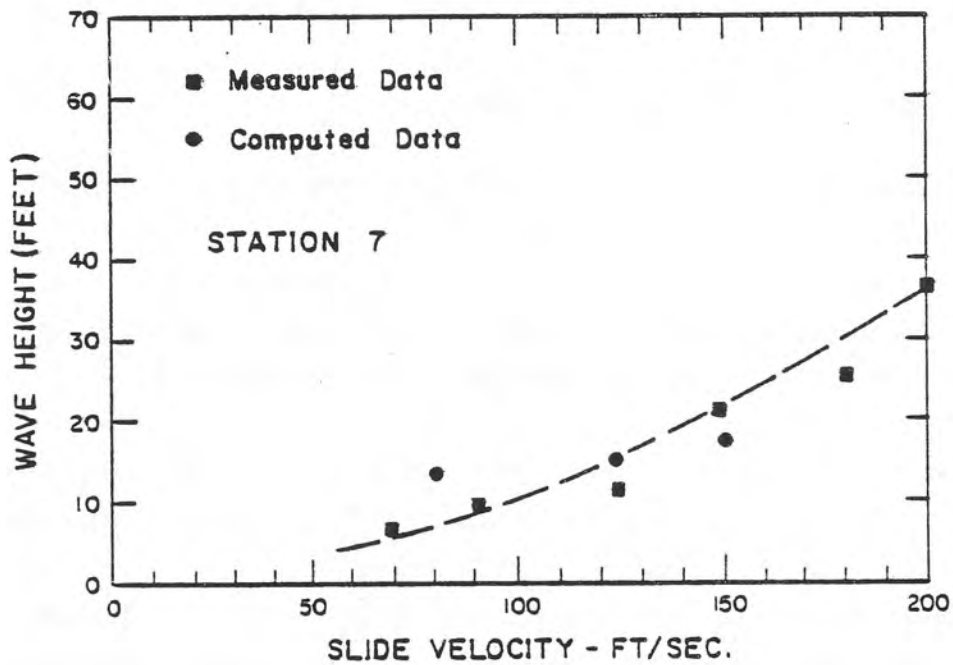
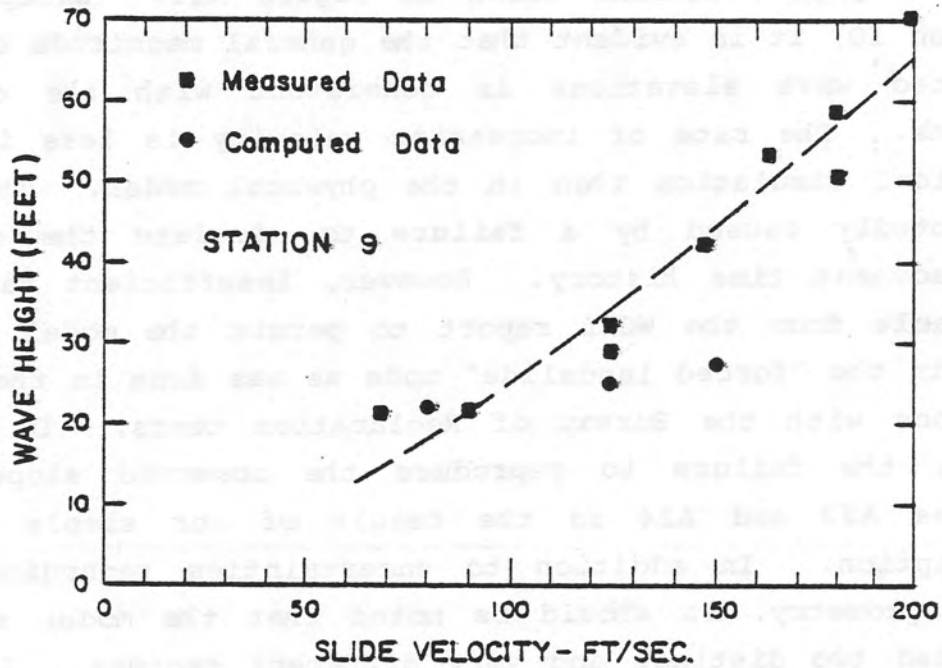


FIGURE A23: Measured and computed elevations of the first wave at Stations 3 and 7.



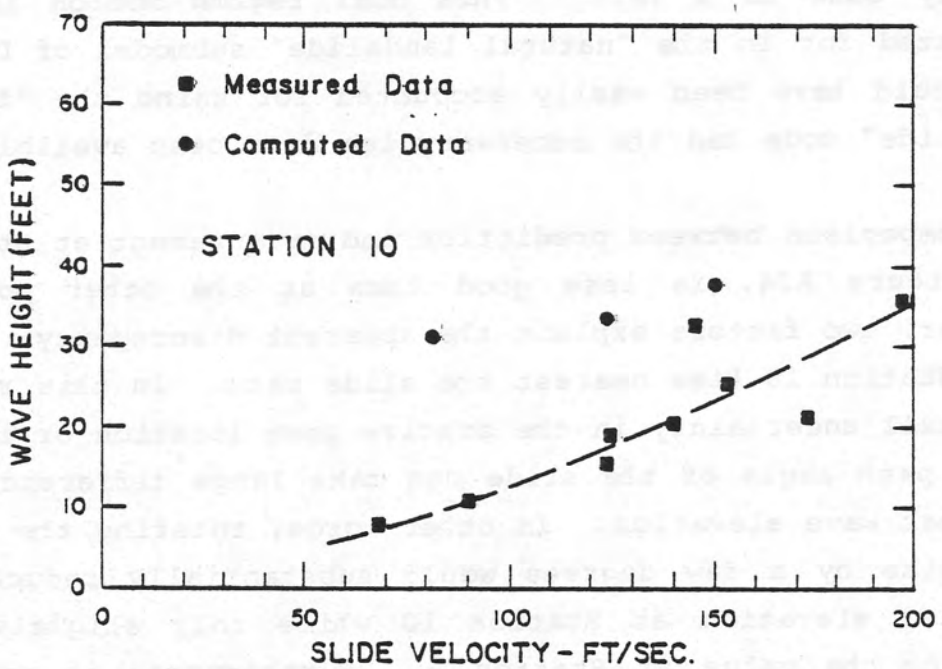


FIGURE A24: Measured and computed elevations of the first wave at Stations 9 and 10. Refer to text for discussion of deviation at Station 10.

for the gage locations shown in Figure A21. Except for Station 10, it is evident that the general magnitude of the computed wave elevations is consistent with the observations. The rate of increasing velocity is less in the numerical simulation than in the physical model. This is undoubtedly caused by a failure to simulate the actual displacement time history. However, insufficient data is available from the WCHL report to permit the model to be used in the "forced landslide" mode as was done in the comparisons with the Bureau of Reclamation tests. In other words, the failure to reproduce the observed slopes in Figures A23 and A24 is the result of our simple slide description. In addition to uncertainties regarding the slide geometry, it should be noted that the model slides included two distinct and very different regimes. In the first, the gravel bags commenced motion on a steep and very smooth sheet metal ramp. However, upon entering the water, the bags immediately encountered the rough model surface and rapidly came to a halt. This dual regime motion is not accounted for in the "natural landslide" submodel of LSWAVE but could have been easily accounted for using the "forced landslide" mode had the necessary lab data been availble.

The comparison between prediction and measurement at Station 10, Figure A24, is less good than at the other points. However, two factors explain the apparent discrepancy. Note that Station 10 lies nearest the slide path. In this region any small uncertainty in the precise gage location or in the exact path angle of the slide can make large differences in the peak wave elevation. In other words, rotating the slide clockwise by a few degrees would substantially reduce the computed elevation at Station 10 while only slightly increasing the value at Station 9. Furthermore, it appears that the numerical grid point chosen to represent Station 10 was actually one or two grid points downstream of the actual location of Station 10.

Despite the greater sophistication of the model slide in Series 2, the lack of wave data prevents any but the simplest comparisons. The shape table adopted in the simulation was as follows:

Little Creek Ridge, Series 2

XSL (Ft)	HSL (Ft)
0	0
65	58.7
195	102.7
325	161.3
455	146.7
585	102.7
715	88.0
845	73.3
975	73.3
1105	51.3
1235	29.3
1300	0

This table conforms to the shape shown in Figure A22; recall that the slide thicknes, HSL, is the vertical thickness, computed here for a ramp slope of 50°. In accordance with findings of the laboratory model, the friction slope was set equal to 0°. That this did indeed produce a slide motion consistent with the physical model was confirmed by the computed slide displacement history which agreed well with WCHL data.

The resulting comparison is a single data point. Table 2, page 37 of the WCHL report (1970) shows peak height versus ramp angle and "Surcharge." Surcharge refers to extra bags plaed on the slide in an effort to achieve the desired displacement history. However, as noted earlier, the tendency of the bags to freeze upon encountering the rough model surface, prevented the desired result.

The comparison between numerical and physical results from the WCHL table is:

Reservoir elevator	=	2475 ft.
Ramp angle	=	50°
Surcharge	=	0
Observed wave at Station 3	=	36 ft.
Computed wave at Station 3	=	31 ft.

This result is entirely satisfactory considering the various uncertainties and approximations involved.

A3.3.3 Additional remarks

It is noted that the case of waves traversing an earth dam is very similar to the problem of waves on a beach. The user of LSWAVE may wish to supplement the model results with considerations obtained from coastal engineering practice.

In particular, LSWAVE might be run first using the actual depths over the dam for input to the depth grid as adopted in the foregoing comparison. Then a second run might be made with an open boundary imposed at the toe of the dam. This would eliminate the dam entirely--values computed at the fictitious open boundary would correspond to the incident wave system arriving at the dam. This artifice might be especially useful in the event that the dam is very small compared with the reservoir in which case adopting a space step small enough to resolve the dam would be excessively costly.

Once the incident wave train is established in this manner, the user should follow the detailed procedures given in Section 7.2, pages 7-15 to 7-59, of the Shore Protection Manual (U.S. Army Coastal Engineering Research Center, 1977). These pages guide the reader in a step-by-step fashion through the computation of wave runup and overtopping for both regular (periodic) and irregular waves.

A3.4 Downstream Flood Routing

When the landslide-generated wave arrives at a dam, it may create extensive overtopping. If the dam does not break, a large short-period flood may be generated downstream. It would be of interest to calculate the discharge over the dam based on the landslide wave model, LSWAVE, developed in this study, and then feed the resulting outpout into an existing flood routing model to estimate the downstream flow.

A search of existing flood routing models was made for this purpose. A kinematic model like the Muskingum routing technique (see, for example, Weinmann and Laurenson, 1979), is simple and economic. However, it has a shortcoming of neglecting the inertial effects of the flow which are important in the propagation of huge flood waves. In order to have a general model which is capable of handling various situations, a model using the St. Venant one-dimensional equations as the basic equations was sought. These partial differential equations, defining non-uniform unsteady, open-channel flow are given in standard reference works (e.g., Gilcrest, 1950; Stoker, 1957; Linsley, et al., 1958; Henderson, 1966). Chow (1959) gives a complete discussion of the equations for conservation of mass and conservation of momentum. The assumptions and restrictions in deriving these equations can be found in Strelkoff (1969) and Chiang (1977). Different types of momentum equations are presented and discussed by Yen (1973) and Jolly and Yevjevich (1974).

Various kinds of numerical models, including those using explicit, implicit, and characteristic methods, have been investigated. In this study, the flood is a short period wave due to huge overtopping above a dam after a landslide occurs upstream of the dam. The effects on flood routing caused by bank storage (Pogge, 1968; Zitta and Wiggert, 1971) and due to aquifer storage (Chiang, 1977) were not considered important here. For the propagation of waves

following the rupture of a dam, the reader is referred to, among others, Su and Barnes (1970), Vasiliev (1971), Sakkas and Strelkoff (1973), Balloffet, et al.(1974), Xanthopoulos and Koutitas (1976), Rajar (1978), and Chen and Armbruster (1980).

In the early 1970's, the National Weather Service Hydrologic Research Laboratory began developing a model called DWOPER (Dynamic Wave Operational Model). It uses a weighted four-point implicit finite difference scheme. The model allows for not only irregular intervals, lateral inflows and off-channel storage, but also backflow effects, variable roughness parameters, wind effects, dendritic river systems, lock and dam conditions, and local head losses such as bridge contraction-expansions (Fread, 1980). It has an automatic calibration feature for determining the optimum roughness coefficient efficiently (Fread and Smith, 1978). Similar to DWOPER, another numerical model named DAMBRK was also developed at the National Weather Service Hydrologic Research Laboratory (Fread and Harbaugh, 1973; Fread, 1980). This model, DAMBRK, is capable of simulating some dam breach problems.

For the purposes of the present study, it was decided to adopt the model DAMBRK developed at NWS to simulate the flood due to huge overtopping generated by landslides, because

(a) DAMBRK is a general model with a variety of features mentioned in the previous paragraph;
(b) it is well documented; and
(c) it is maintained by NWS and is open to the public.

Both LSWAVE and DAMBRK require large computer storage. Consequently, it is not wise to combine these two models and run them at the same time because there is no feedback information from DAMBRK to LSWAVE. All input data for DAMBRK

are independent of the data in LSWAVE except the upstream hydrograph which is the discharge hydrograph over the dam produced by LSWAVE.

Furthermore, most of the time it is not required to run the downstream flood routing model when simulating landslide-generated waves. Therefore, it was decided to separate the downstream model from LSWAVE. When LSWAVE is run, the discharge hydrograph over the dam is stored into a file. This file can then be used to supply the upstream boundary condition to DAMBRK. When testing one of these two models, it is not necessary to run the other one.

Version B, which is a short version of version A of DAMBRK, is used. In order to utilize this computer program (DAMBRK, version B, dated September 10, 1980), it was necessary to make the following modifications to the program:

(a) Create an option of reading hydrograph data from either card deck or other data files;

(b) Increase storage space for the hydrograph data;

(c) Provide computer coding to read a hydrograph from the data file created by LSWAVE and convert the data to a hydrograph suitable to DAMBRK;

(d) Change some conditional branch statements and some formats for writing and plotting data because the time scale in the present study is much smaller than the originally designed time scale; and

(e) Correct a few errors.

Furthermore, a few read statements using special features in an IBM computer were modified such that the program can be run on a CDC computer as required under the present contract. The modifications to the original program are summarized in Appendix B.1 of Part B.

Overtopping data computed by the numerical model LSWAVE, for slide A (run number SLA91), were stored and then fed into the modified DAMBRK as the input hydrograph. The tranfer of data from LSWAVE to DAMBRK was successful. The modified DAMBRK read the discharge hydrograph from the stored file and other data from the card deck, and then continued the simulation process. The propagation of a huge wave due to overtopping was simulated through the 0.7 miles reach (Gunnison River) downstream from Morrow Point Dam. Based on testing of several combinations of input data, the following conclusions were drawn:

(a) Although DAMBRK has been tested successfully for the case of Teton Dam flood on June 5, 1976, and other dam breach problems, it did not perform adequately for the extreme flood generated in this study. The Teton Dam flood has a peak discharge of 1,652,300 cfs and a time to peak (i.e., duration of rising stage) of 1.25 hrs as computed from DAMBRK (Fread, 1980). The overtopping computed from LSWAVE, for the case of slide A, has a peak discharge of 1,694,000 cfs and a time to peak of about 8 secs. This huge wave of short period is extremely difficult to simulate numerically.

(b) The state of flow in the channel changes from subcritical to supercritical, and back to subcritical during the flood period. Initially, the flow is subcritical on the mild slope channel. When the huge wave arrives, the water velocity is so high that the Froude number exceeds one and the flow becomes supercritical.

(c) The model DAMBRK can handle the cases of (i) subcritical flow, (ii) supercritical flow through the entire channel reach, or (iii) supercritical flow for an upstream portion, and subcritical flow for the remaining downstream portion. The state of any portion cannot be changed during a simulation run. If it is originally a subcritical flow, it can not be changed to a supercritical flow and vice versa.

(d) To let it run under the condition of subcritical flow, either the Manning coefficient has to be unreasonably high, or the discharge has to be low. For the high discharge in this study, the Manning coefficient has to be as high as 0.5. The results are then meaningless because flow behaves like a mud flow. For a Manning coefficient of 0.055, DAMBRK was repeatedly tested by halving the ordinate of the input hydrograph each time. It could successfully handle the case of peak discharge of 11666 cfs for the topographic condition set up for Gunnison River, 0.7 miles downstream from Morrow Point Dam. The Froude number became too high and the simulation was forced to stop when the peak discharge was higher than 23290 cfs with a time to peak of about 12 sec.

(e) To let DAMBRK run under the condition of supercritical flow in a channel of mild slope, the initial discharge has to be high enough to obtain a high Froude number. For the Gunnison River tests in this study, the simulation succeeded when the minimum discharge everywhere was maintained at a high value of 60,000 cfs; however, it failed when the initial discharge was 50,000 cfs. For the case of such an unreasonably high base flow of 60,000 cfs, the waves propagated too fast and the peak heights at downstream stations were overestimated.

In summary, the transition of hydrograph data from the developed model LSWAVE to a downstream flood routing model was performed successfully. The flood routing model DAMBRK did not respond well for this case of huge waves of short wave period on a mild slope channel, because the flow changed from a subcritical to a supercritical state. An appropriate numerical model capable of simulating this transition from subcritical flow to supercritical flow has not been found in the literature. It is suggested that an appropriate model could be developed using DAMBRK or some other existing model as a basis.

A4.0 CONCLUSIONS AND RECOMMENDATIONS

A numerical model (LSWAVE) has been developed to simulate the generation, propagation, and dissipation of landslide-generated waves. The model has been tested for the case of waves generated by three hypothetical landslides in the Morrow Point Reservoir area. The results compare favorably with laboratory measurements. When tested for the case of the Lituy Bay rockslide, of July 9, 1958, the computed wave heights, based on a coarse computational grid, are satisfactory when compared to the field observations after the event. The model has also been satisfactorily tested for the case of a landslide into Mica Reservoir, the results generally agreeing with laboratory data for this earth dam. The time history of discharge over the dam is stored as an output file which is fed to a downstream flood routing model.

An area meriting additional effort is development of a more appropriate downstream flood routing model capable of handling the extreme overtopping rates which may result from a landslide generated wave. Combined with LSWAVE, such a model would provide the user with a comprehensive package for wave-related dam safety analysis. Also of interest, but not addressed here, are the problems of dynamic wave loading on the dam structure including the problem of wave impact forces.

APPENDIX A1. LIST OF NOTATIONS

A = cross-sectional area of landslide

a = horizontal acceleration of landslide

a_s = acceleration of landslide along the direction of the slope

A_p = projected area of the submerged part of landslide, on a plane normal to the direction of slide

A_s = horizontal area of projection of the submerged part of landslide

A_v = projected area of the submerged part of landslide on a plane parallel to the local bottom surface

C = wave celerity

C' = a constant equal to $2n^2g/(1.486)^2$

C_c = Chezy coefficient

C_D = drag coefficient

C_d = discharge coefficient

C'_d = a dimensionless discharge coefficient

C_{Dp} = pressure drag coefficient

C_{Dv} = viscous drag coefficient

C_{ED} = energy dissipation coefficient

D = water depth relative to initial water surface

D_p = pressure drag (form drag)

D_v = viscous drag (surface drag, skin drag)

F = total horizontal resistance due to water on landslide

F_x = total resistance along x-direction

F_y = Total resistance along y-direction

g = gravitational acceleration

H = $\eta+D$, total water depth relative to instantaneous water surface

HSL = height (thickness) of landslide as a function of XSL

$(HSL)_{max}$ = maximum submerged height (thickness) of landslide

H_t = total head above a dam

i = index of landslide segments

K = added mass coefficient

n = Manning's friction coefficient

q = discharge over a unit width of a dam

R = hydraulic radius

S_{bx} = x-component of bottom stress

t = time

TSL = time scale of landslide motion

U = depth averaged water velocity along x-direction

$\overline{U}$ = U averaged over four neighboring grid points

V = depth averaged water velocity along y-direction

$\overline{V}$ = V averaged over four neighboring grid points

V_h = horizontal velocity of landslide

V_i = normal water velocity at a half grid spacing inside a boundary

V_o = normal water velocity at a half grid spacing outside a boundary

V_o^P = V_o under a partial reflection condition

V_o^R = V_o under a total reflection condition

V_o^T = V_o under a total transmission condition

V_r = velocity of landslide relative to far field water motion

V_s = volume of landslide

V_s' = submerged volume of landslide

VSL = horizontal velocity of landslide as a function of TSL

W = width of landslide

x = horizontal coordinate

XSL = horizontal distance along landslide, measured from leading edge

y = horizontal coordinate

α = inclination, angle between the instantaneous slide direction and the horizontal plane

γ = a delta function, equals 1 only if its associated term is applied to the first element of landslide

δ = a delta function, equals 0 when landslide is at point of interest, otherwise, 1

η = surface elevation relative to initial water surface

ϕ = friction angle

θ = forward direction of landslide motion, measured counter-clockwise from x-axis

ρ = unit mass of water

ρ_s = unit mass of slide material

PART B

USER'S GUIDE

B1.0 INTRODUCTION

This section of the report presents a detailed discussion of the design and use of program LSWAVE. The core of this numerical model is a hydrodynamic submodel which simulates the propagation of waves generated by other sub-routines. There are three wave generating submodels built into LSWAVE. The first submodel is the basic one. Its function is to generate the wave caused by a "forced" landslide, so called because the motion of the landslide is forced to follow a predetermined time-history. The magnitude of the generated wave is primarily a function of the rate of volume intrusion. The second submodel, the "natural" landslide sub-model, computes the motion of a slide based on the initial position, size, and shape of the slide mass, topography, and other physical parameters. The user does not have direct control on the intrusion rate of the slide. The third sub-model is a minor one. It produces a wave associated with a simple bottom up-lift which can occur near a fault line. There is room for the user to include other wave-generating submodels according to his needs.

The remainder of this section is arranged as follows. Chapter B2 presents the general concepts of the numerical model. The ways to describe various boundaries are emphasized. Chapters B3, B4, and B5 describe the input parameters required by the three wave-generating submodels. A code, LSKEY, allows one to choose the "forced" landslide,

the "natural" landslide, or a fault induced bottom motion as the forcing mechanism.

Chapter B6 presents a detailed description of the input parameters for the model. An index of the parameters is attached at the end of that chapter. Chapter B7 briefly describes various output data from the model. Chapter B8 contains a listing of the whole computer code. A set of sample input data and the partial output based on this sample input are presented in Chapter B9.

B2.0 HYDRODYNAMIC MODEL

B2.1 General Description

The numerical model uses a finite difference method to compute the spatial and temporal variations of water surface elevation. The first task before running the model is to establish the computational grid for the area of interest. To simulate the waves generated by a "natural" landslide, not a "forced" landslide, the landslide area must be included in the computational mesh. Figure B1 depicts an example layout of the numerical model for a reservoir.

A right-hand Cartesian coordinate is chosen such that x increases from top to bottom, and y increases from left to right, as indicated in Figure B1. This is the convention used in computer printouts. Any point with indices (i,j) is at a distance (i x (DX), j x (DY)) away from the origin. The symbols (DX) and (DY) denote uniform grid spacings along the x- and y-directions, respectively. The index i ranges from 1 to M, and j from 1 to N. The total number of grid points is (M x N) which is limited to 2,000 in the present version of the model. Most grid points outside the reservoir boundary are not referenced during the computation. Grid points defined with positive depth values (to be discussed later) and their neighboring points are called computational points. The total number of computational points is limited to 1350. For the example reservoir shown in Figure B2, the values of M and N are 15 and 64 respectively, and there are about four hundred computational points.

A space-staggered scheme is adopted to increase the efficiency of programming. Surface elevation and two components of velocities are described at different grid points as

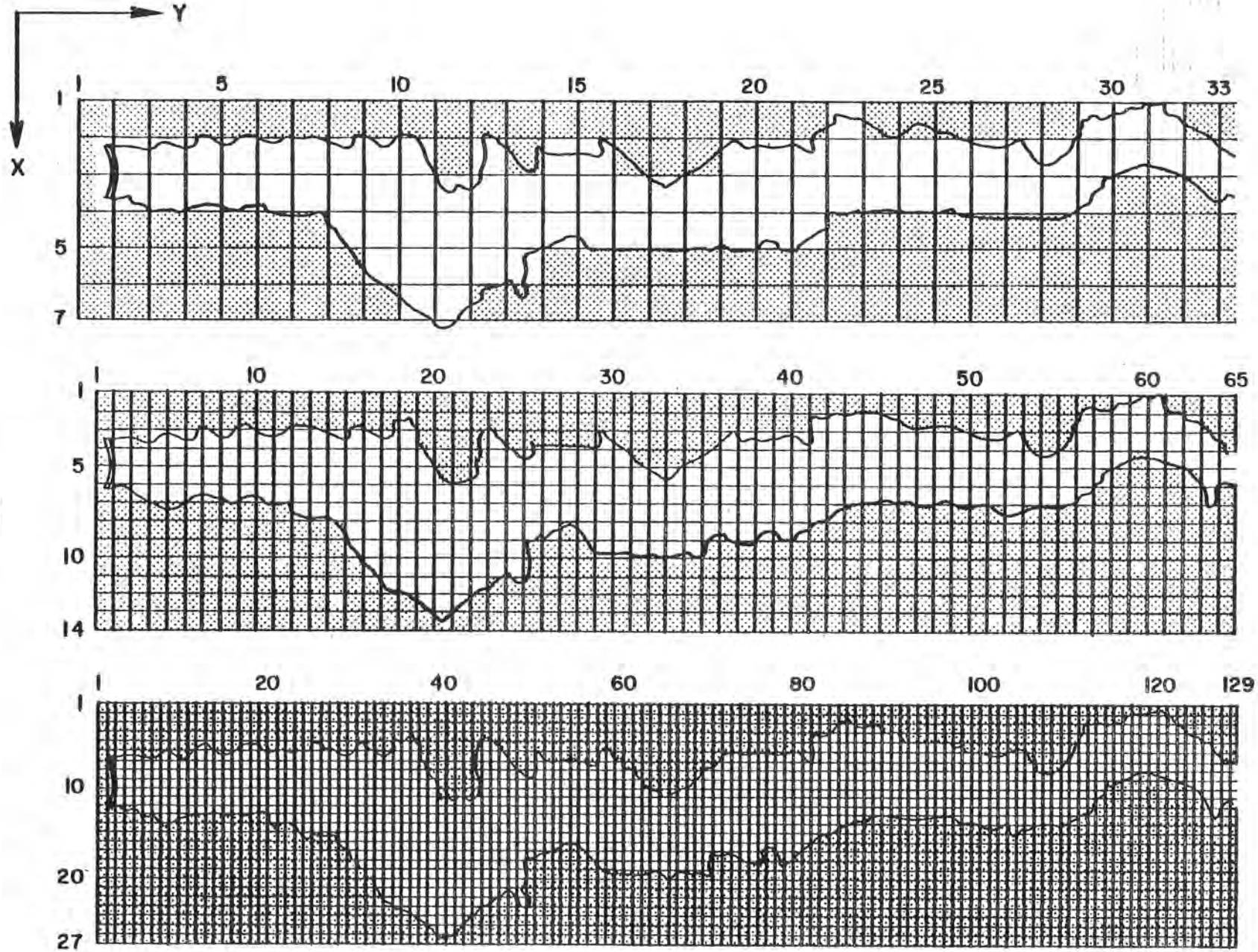


FIGURE B1: Finite-difference representation of a reservoir with the grid spacing equal to, from top to bottom, 500 feet, 250 feet, and 125 feet.

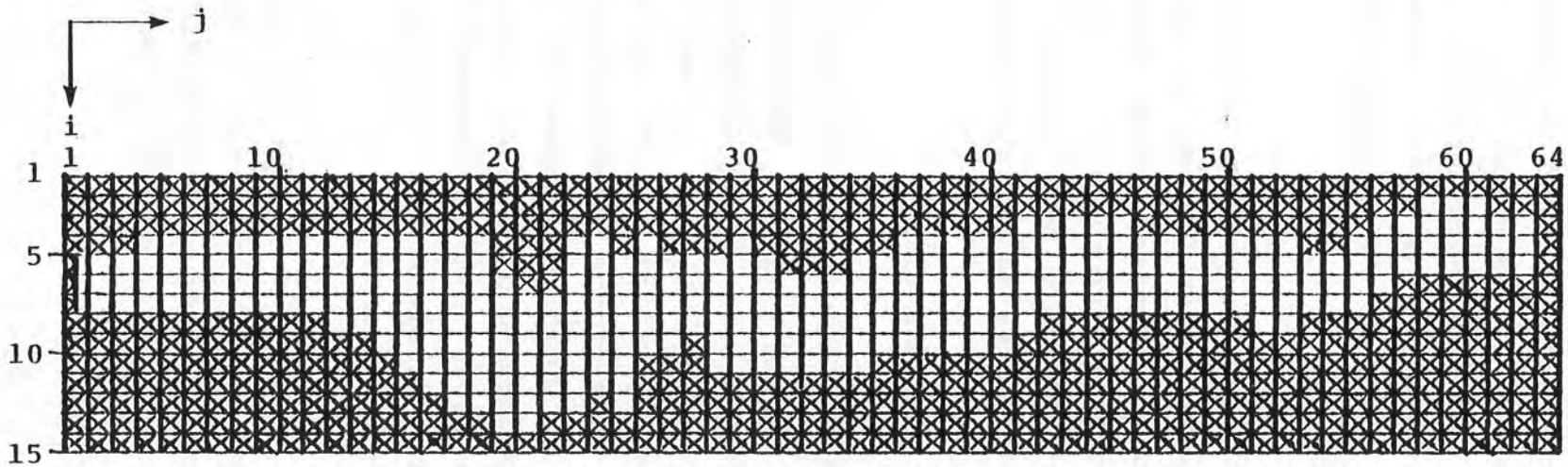


FIGURE B2: Finite-difference representation of a reservoir, with the elevation points falling on intersections of the grid system.

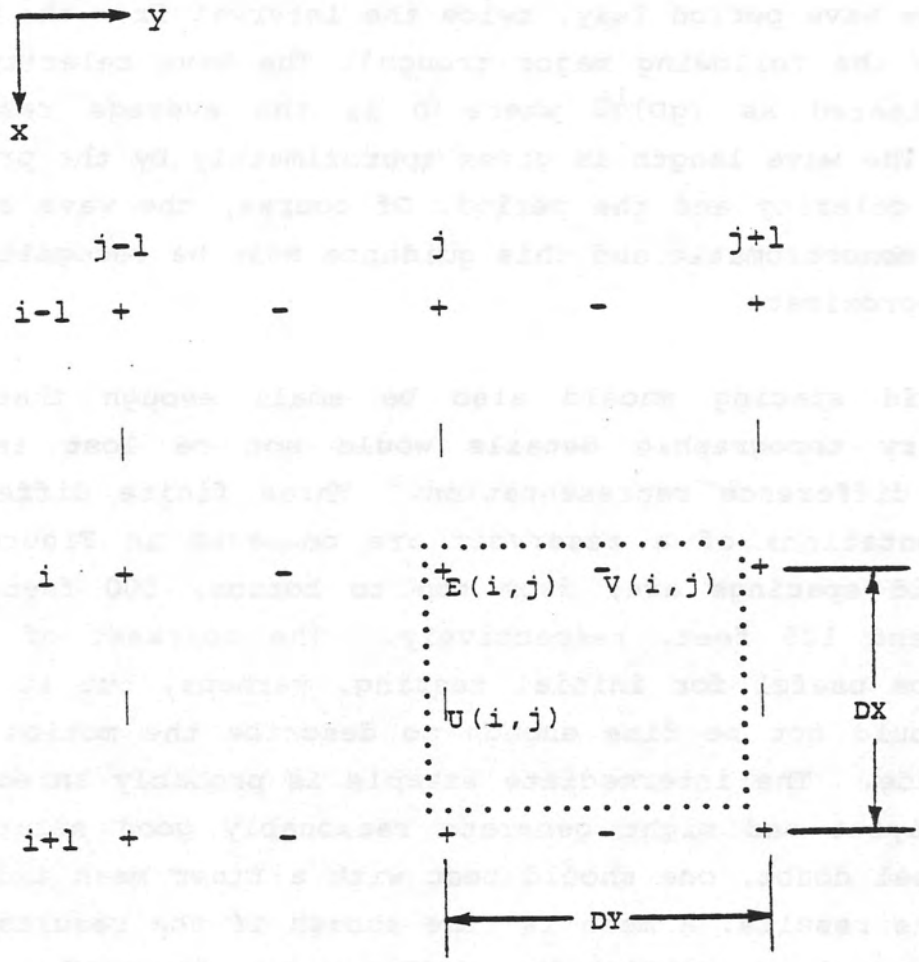
indicated in Figure B3. Depth and friction coefficient are defined at the elevation points. All variables in the small square surrounded by the dotted lines in Figure B3 are assigned the same indices (i,j). Symbols U(i,j) and V(i,j), as used in the computer program, are actually defined at ((i+½)(DX),j(DY)) and ((i(DV),(j+½)(DY)), respectively. The value of a symbol at a point is the averaged value for the grid centered at that point.

Keeping Figure A3 in mind, the user may construct the grid mesh either with elevation points falling at the center of the squares, or with the elevation points falling on the intersections. Figures A1 and A2 are plotted with the elevation points at intersections which seem to make it somewhat easier to visualize the boundary as defined in this model.

B2.2 GRID SPACINGS AND TIME STEP

The grid spacings mentioned here are those for the hydrodynamic submodel. The sub-spacings in the landslide area will be discussed in Chapter B3. In the hydrodynamic model, different grid spacings, DX and DY, are allowed for the two directions. It is recommended to set DY the same as, or close to, DX in order to minimize computational error. A smaller grid spacing will naturally improve the description of wave generation and propagation. However, one should note that the computational cost is inversely related to the cube of the step size in this explicit scheme. One must therefore balance resolution against economy.

If the wave length can be estimated, the grid spacings can be adjusted to adequately resolve the wave. A minimum of 6 space steps per wave length is required to obtain reasonable accuracy. If possible, it is recommended to have a minimum



LEGEND:

+: Point of definition of surface elevation (E), depth (D), and friction coefficient (C); also called "elevation point" in this study

|: Point of definition of x-direction velocity (U)

-: Point of definition of y-direction velocity (V)

FIGURE B3: Definition sketch of space-staggered scheme.

of 10 space steps per wave length. The wave length can be estimated by performing preliminary calculations to determine the wave period (say, twice the interval from the major peak to the following major trough). The Wave celerity can be estimated as $(gD)^{1/2}$ where D is the average reservoir depth. The wave length is given approximately by the product of the celerity and the period. Of course, the wave system is not monochromatic and this guidance must be recognized as very approximate.

The grid spacing should also be small enough that any necessary topographic details would not be lost in the finite difference representation. Three finite difference representations of a reservoir are compared in Figure B1. The grid spacings are, from top to bottom, 500 feet, 250 feet, and 125 feet, respectively. The coarsest of these would be useful for initial testing, perhaps, but it probably would not be fine enough to describe the motion of a landslide. The intermediate example is probably an economical layout and might generate reasonably good solutions. To dispel doubt, one should test with a finer mesh and compare the results. A mesh is fine enough if the results from this mesh have relatively small errors compared to the results from a mesh of half the step size. Notice that the results are directly comparable only when the input parameter CED (to be discussed later) is set to either 0 or 1 because the effects of CED depend to some degree on the grid spacing for values between 0 and 1.

The grid spacing should be smaller than the dimension of the landslide. In other words, a landslide should cover several grid points at any instant. It is suggested that a landslide be wider than three or four grid spacings.

If the discharge over a dam is of interest, the grid spacing should be adjusted so that the crest length of the dam is an integral multiple of the step size.

Because this model adopts an explicit finite difference scheme, the computational time step, DT, is limited by the Courant stability criterion

$$((DT)(((DX)^2+(DY)^2)(G)(Dmax))^{\frac{1}{2}}) \text{ .LE. } ((DX)(DY)) \qquad (B1)$$

where G is the gravitational acceleration, and Dmax denotes the maximum water depth in the computational region. When the wave height is large relative to the water depth, the left hand side of the inequality above should be multiplied by

$$1 + Emax/(Dmax+Emax)$$

where Emax is the maximum possible wave height, and for safety assumed that Emax will occur at the same point as Dmax.

Better accuracy will be obtained if the computational time step, DT, is fine enough to describe the landslide motion smoothly. It is recommended to have DT smaller than one twentieth of a typical wave period, if the period can be estimated. A smaller DT gives a larger safety factor against numerical instability. The computational cost is approximately inversely proportional to DT.

B2.3 Input Data for Water Depth

Either bottom elevations or water depths are defined at elevation points (points marked by "+" in Figure B3) through input data. A bottom elevation is defined positive upward from a datum. A depth is defined positive downward from the initial still water surface. A variable, Z0, which denotes the initial elevation of the water surface, controls whether the input data are elevations or depths. If the value of this variable is zero, the input data, D(K), are treated as water depths. If Z0 is not zero, the input elevations, D(K), will be converted to depths by:

$$\text{Depth}(K) = 0 \qquad \text{if } D(K) = 0 \tag{B2}$$

and

$$\text{Depth}(K) = Z0 - D(K) \qquad \text{if } D(K) \text{ .NE. } 0 \tag{B3}$$

A conversion factor, DCONV, can be input to convert the input depths from any convenient unit system to the unit system used in the model. After the conversion, all variables should use FPS, CGS, or MKS units consistently. The only exception is Manning's n, which should always be input in the FPS system.

The depth values should be defined at every elevation point which is covered by water. The value at a point represents the average depth over the whole grid which is centered at that point. In the case of a "natural" landslide, which will be discussed later, the depth should also be defined at all points traversed by the landslide even if outside (above) the reservoir. It is optional whether to define the depth or leave it zero for those points which are not involved in the computation. The depth for a point outside the water body has a negative value. The water motion is

limited to the region where the depths are positive. The depths at those elevation points immediately outside any kind of boundary (model boundary, as shown in Figure B3, not real boundary) should be zero or negative, otherwise that boundary element will be invisible to the model. In other words, the model "sees" a boundary when the depth changes from positive to either zero or negative across grid points.

The land elevation at the landslide area is defined as the elevation of the slip surface of the slide. In other words, the slide is treated as a moving mass riding atop the specified land elevation.

In order to increase computational efficiency, the depth value along all grid boundaries, $i = 1$, $i = M$, $j = 1$, and $j = N$, should be either zero or negative. If a computational point, where the depth is positive, falls on one of those two extreme rows or two extreme columns, the computation will stop with an error message and the user should either change that depth value, or increase the number of rows or columns. The total numbers of rows and columns are M and N, respectively. In Figure B3, M and N are 15 and 64 respectively. Depths are defined on and inside boundaries. Islands are allowed to exist inside the computational domain.

B2.4 Boundaries

There are three kinds of computational boundaries: dam, open boundary, and partial-reflection boundary.

B2.4.1 Dam

A dam is located across velocity points as illustrated in Figure B4. Figure B4(a) shows a simple dam with a width covering three grid points. If this is the only dam in the model, the input parameters MWEIR, number of dam segments normal to the x-axis, is zero; NWEIR, the number of dam segments normal to the y-axis, is 1; JWEIR(1), the column number of the first dam segment, is j; I1WEIR(1), the starting row number of the first dam segment, is i; and I2WEIR(1), the ending row number of the first dam segment, is i+2. The dam is recognized by the model as having a width of 2(DX), from $(i, j-\frac{1}{2})$ to $(i+2, j-\frac{1}{2})$. Notice that JWEIR(K), as well as IWEIR(K), which is the row number of the Kth dam segment normal to the x-axis, is defined as the column number (or row number) for the neighboring elevation point which is located <u>inside</u> the water region. Suppose column j is in a dry area, and column j-1 is the water region, then JWEIR(1) should be j-1 instead of j.

Figure B4(b) illustrates two dams. The one on the left hand side could occur when a convex dam is associated with relatively small grid spacings. It contains two segments which are normal to the x-axis, and three segments normal to the y-axis. The dam on the right hand side is the model representation of a straight dam located diagonally across the grid mesh. The input data to describe these two dams are illustrated in Table B1. The freeboards HWEIRI(K) and HWEIRJ(k) are defined relative to the initial water surface.

The complicated dam systems in Figure B4(b) yield more error than the simple dam in Figure B4(a). It is better to adjust the orientation of the grid mesh to simplify the representation of a dam. Both MWEIR and NWEIR are limited to 9.

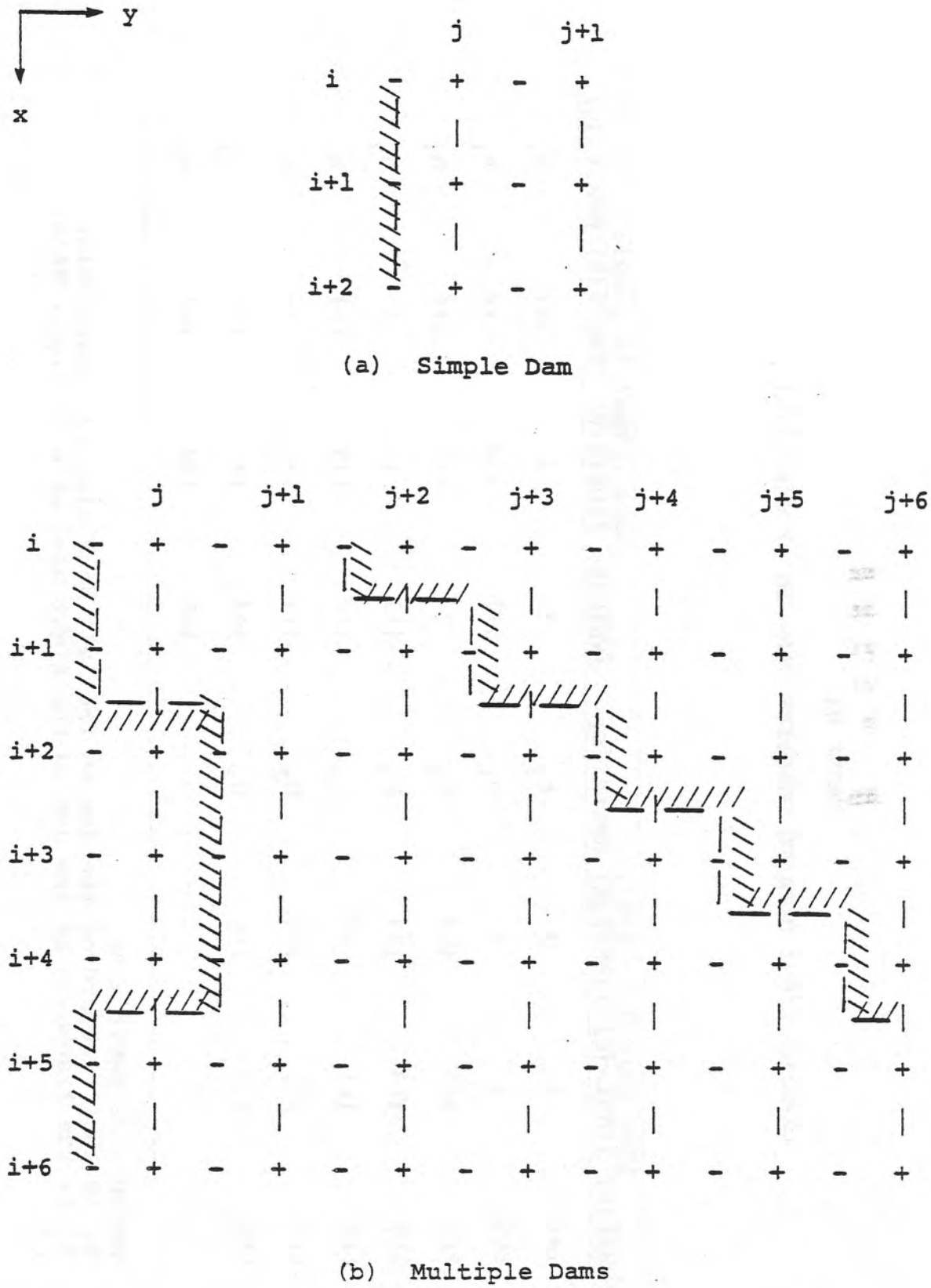


FIGURE B4: Definition sketch for the location of dams.

TABLE B1

EXAMPLE INPUT DATA TO DESCRIBE DAMS IN FIGURE B4(b)

	Dams normal to x-axis				Dams normal to y-axis			
K	IWEIR(K)	J1WEIR(K)	J2WEIR(K)	HWEIRI(K)	JWEIR(K)	I1WEIR(K)	I2WEIR(K)	HWEIRJ(K)
1	i+1	j	j	h_1	j	i	i+1	h_1
2	i+5	j	j	h_1	j+1	i+2	i+4	h_1
3	i+1	j+2	j+2	h_2	j	i+5	i+6	h_1
4	i+2	j+3	j+3	h_2	j+1	i	i	h_2
5	i+3	j+4	j+4	h_2	j+2	i+1	i+1	h_2
6	i+4	j+5	j+5	h_2	j+3	i+2	i+2	h_2
7	i+5	j+6	j+6	h_2	j+4	i+3	i+3	h_2
8					j+5	i+4	i+4	h_2

Note: MWEIR = 7; NWEIR = 8;
h_1 is the freeboard of the dam on the left hand side of Figure B4(b);
h_2 is the freeboard of the dam on the right hand side of Figure B4(b).

B2.4.2 Open boundary

An open boundary is an artificial boundary inside a water region. The open boundary in this model is a transmissive boundary where waves propagate outward freely. It can be used to truncate a computational domain when the water outside this domain has negligible effects on the computational results of interest.

An open boundary is defined along a the line connecting elevation points, whereas a dam is defined along a line connecting velocity points (see Figure B5). Input variables MOB and NOB are the number of open boundary segments normal to the x- and y-axes, respectively. These variables are each limited to 4.

Since the outgoing wave condition used for the treatment of an open boundary in this study is based on a simple wave equation which does not consider the variation of bottom elevation, the bottom elevation at a point on the open boundary is better artificially set to be the same as that of the neighboring inside point, However, this is not usually critical.

B2.4.3 Partially reflective boundary

The water-land boundary is a partially reflective boundary. In this model, the user is not required to define the location of this sort of boundary, since any boundary other than a dam or open boundary is automatically treated as a partially reflective boundary allowing the wave to be partially transmitted. A coefficient, called the energy dissipation coefficient and denoted CED in this report, controls the degree of transmission. The coefficient varies from 0 to 1,

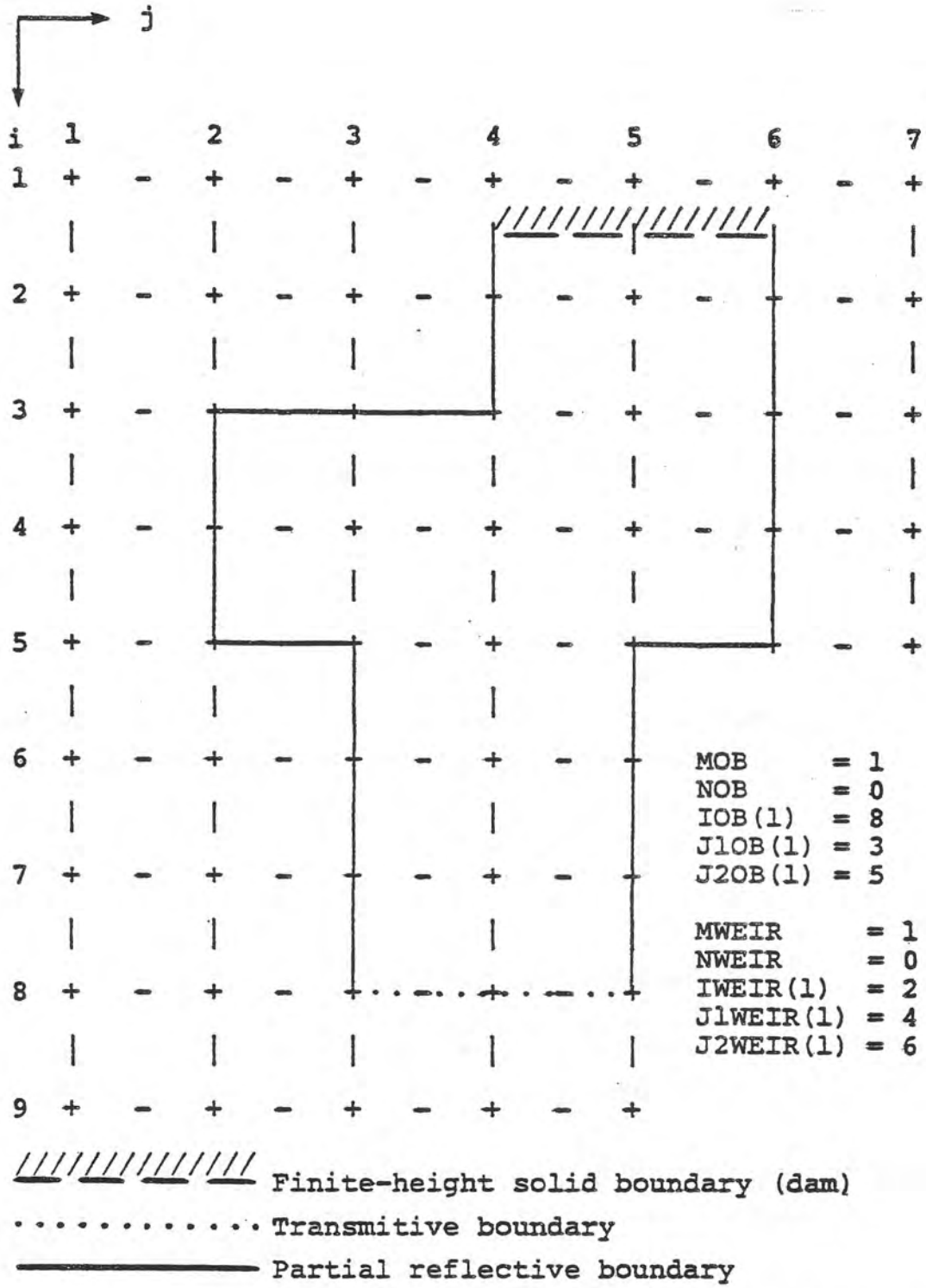


FIGURE B5: Definition sketch of various kinds of boundaries.

with 0 corresponding to the condition of total reflection, and 1 corresponding to the condition of total transmission. One constant value applies to the entire time-space domain.

The coefficient is indirectly proportional to the commonly used "transmission coefficient." The relationship between these two coefficients depends on the characteristics of the wave under study and on some parameters of the numerical model. An appropriate value of the coefficient can be calibrated against theoretical, laberatores, or field data.

For example, Figure B6 illustrates the relationship between this "energy dissipation coefficient" and "wave transmission coefficient" for a particular case in which a sinusoidal standing wave oscillates in a one-dimensional basin of a constant depth of 50 meters, and a length of one half of the wave length of 1680 meters. The basin is closed by partial reflective boundaries at both ends. The grid spacing is 70 meters, and the computational time step is 3 seconds. Amplitude decay for four values of CED was shown in Part A of this report (Figure A3).

The comparisons between laboratory and numerical results for the Morrow Point tests shown in Part A of this report, adopted a value of CED = 0.7 for slide A and a value of 0.6 for slides B and C, which appeared to give good overall agreement. Of course, individual tests might be better approximated using some other value. For example, consider the case of Morrow Point simulation SA121 (Pugh & Harris, 1982). The measured peak height at the dam was about 53 feet whereas the numerical simulation using CED = 0.7 gave about 39 feet.

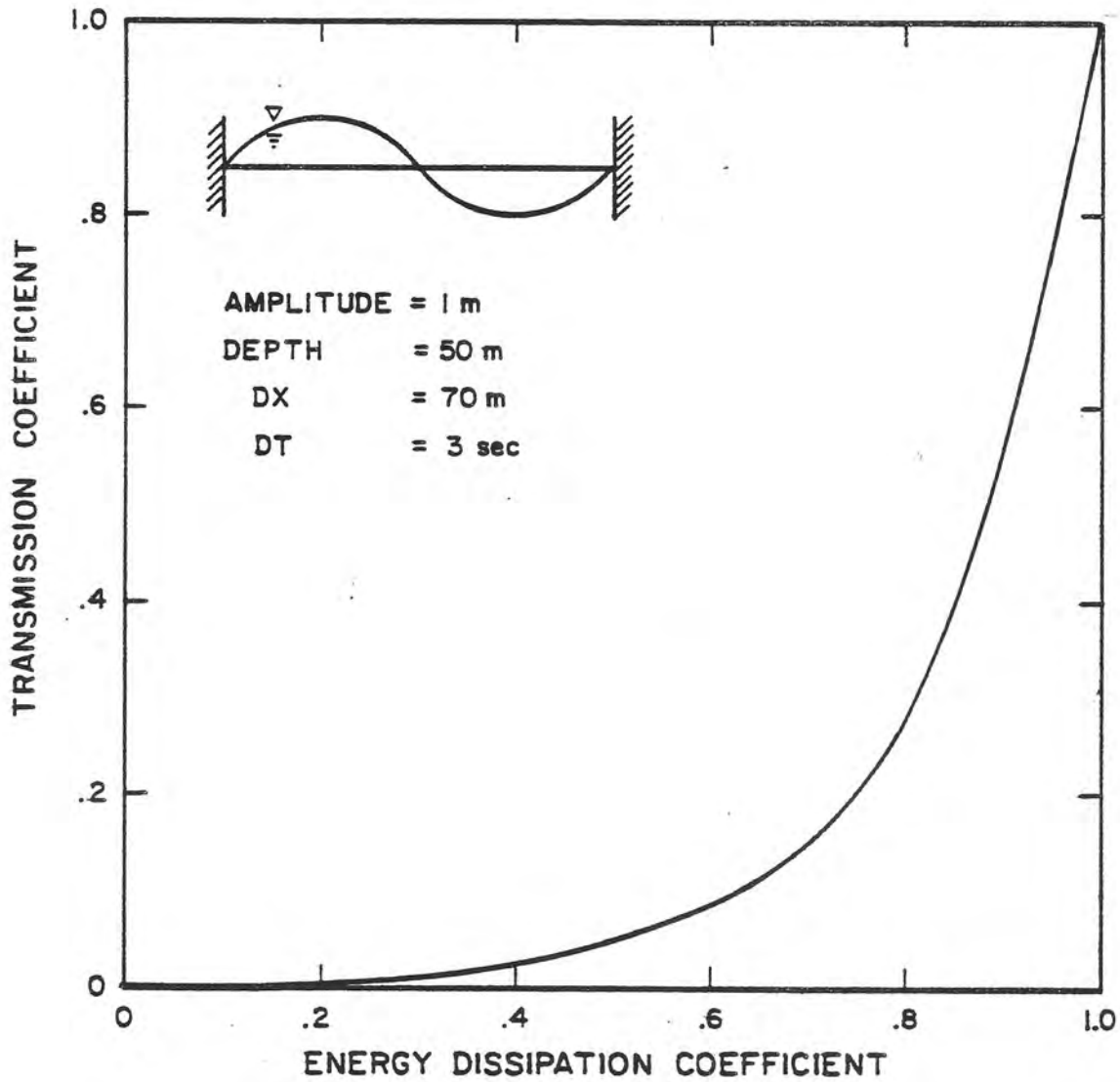


FIGURE B6: Relationship between the conventional transmission coefficient and the energy dissipation coefficient used in this model for a particular case of a simple wave oscillation in a 1-D basin.

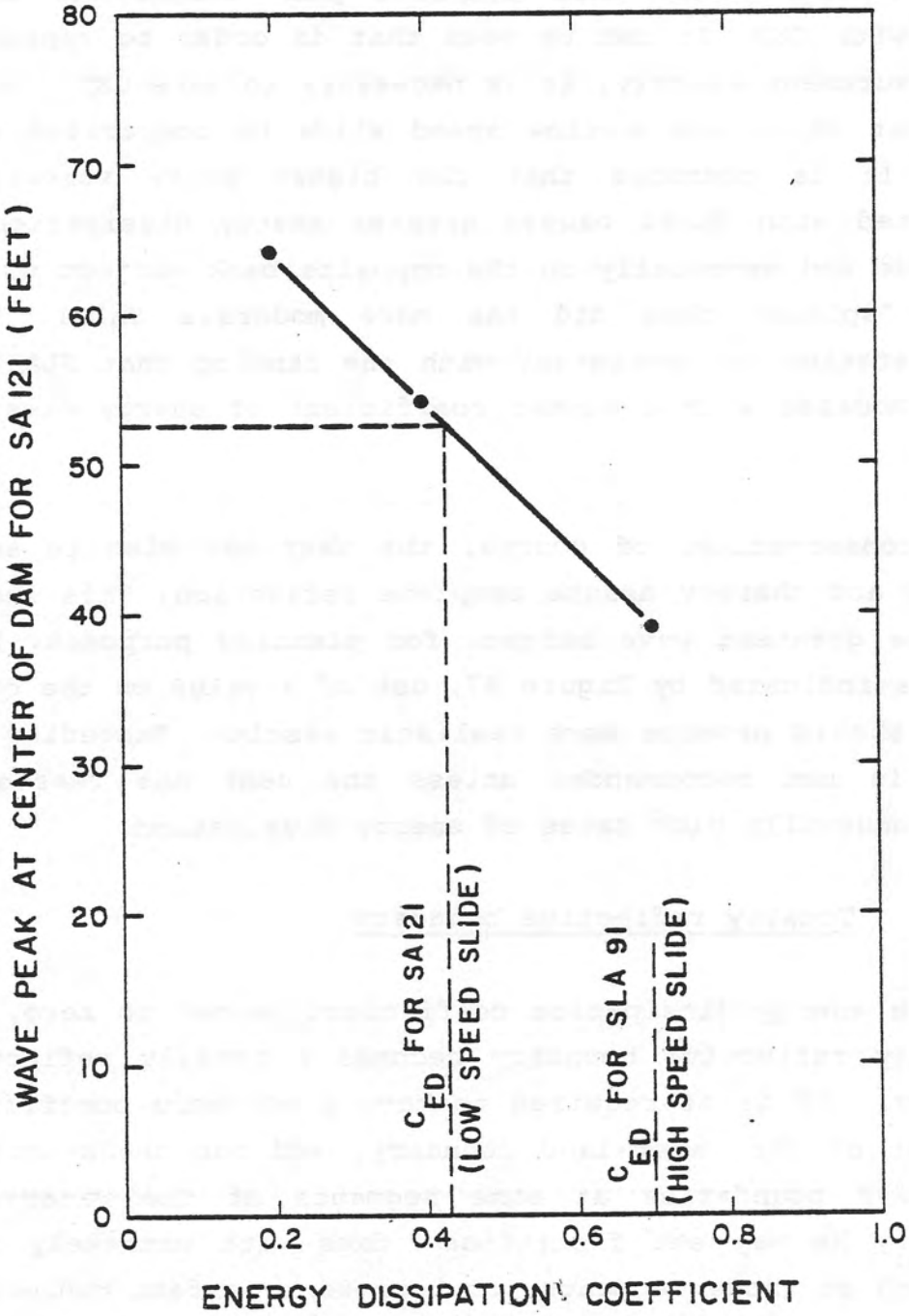


FIGURE B7: Dependence of SA121 peak height at Morrow Point Dam on energy dissipation coefficient.

Figure B7 shows how this computed peak height for SA121 varies with CED. It can be seen that in order to reproduce the measurement exactly, it is necessary to take CED 0.43. Note that SA121 was a slow speed slide by comparison with SLA91. It is probable that the higher entry velocities associated with SLA91 caused greater energy dissipation on the slide and especially on the opposite bank subject to the direct "splash" than did the more moderate SA121. This interpretation is consistent with the finding that SLA91 is better modeled with a higher coefficient of energy dissipation.

To be conservative, of course, the user may wish to adopt CED = 0 and thereby assume complete reflection. This should give the greatest wave heights for planning purposes. However, as indicated by Figure B7, use of a value on the order of 0.5 should produce more realistic results. Exceeding CED = 0.8 is not recommended unless the user has reason to expect unusually high rates of energy dissipation.

B2.4.4 Totally reflective boundary

When the energy dissipation coefficient is set to zero, the partially reflective boundary becomes a totally reflective boundary. If it is required to have a non-zero coefficient for most of the water-land boundary, and one needs totally reflective boundaries at some segments of the water-land boundary, he may set fictitious dams with extremely high freeboard at those segments, to represent perfect reflective walls. A dam in this model is nothing but a finite-height solid-wall.

Figure B5 illustrates the input data to specify the boundaries of a M x N = 9 x 7 basin. There is a dam along i = 2 and an open boundary along i = 8. The depth should be specified at all interior points and boundary points. The depth should be zero along i = 1, i = 9, j = 1, and j = 7, in this example.

B2.5 Discharge Coefficient

The discharge over a dam is computed through the weir formula:

$$q = C_d' \, H_t \, (2gH_t)^{1/2}$$

$$= C_d \, H_t^{3/2} \qquad \text{(B4)}$$

where $$C_d = C_d' \, (2g)^{1/2} \qquad \text{(B5)}$$

is the discharge coefficient used in this model, q is the discharge over a unit width, C_d is a dimensionless discharge coefficient, H_t is the total head above the dam measured at a distance (more than 2.5 H_t) upstream from the dam, and g is the gravitational acceleration.

The discharge coefficient for overtopping due to landslide-generated waves is difficult to determine. It depends on the characteristics of the dam, the water depth, the particle velocity, and the characteristics of the wave on top of the dam. There is no calibrated data base for overtopping caused by huge waves. The calibrated data of formulas discussed in the literature (e.g. Rehbock, 1929; Kandaswamy

and Rouse, 1957; see also U.S. Department of the Interior, 1948, for bibliographies) are all based on studies of steady flows. Therefore, this coefficient should be calibrated through the numerical model using it, if field data or laboratory data are available.

Constants, formulas, tables, or figures such as Figure B8 may be obtained from standard hydraulic textbooks such as Chow (1959, 1964), Henderson (1966), Brater and King (1976), and Merritt (1976). The application of this information to a real case with a huge wave may result in an error since this data is based on idealized cases.

Horton (1907) and Bradley (1952) both present the discharge coefficient for some actual dams. Some values of C_d are higher than 4.0, in English units, even under the normal conditions (see Horton, 1907, p. 132; Kirkpatrick, 1957, pp. 194-208; Chow, 1959, p. 366).

If the user specifies a negative value for C_d, the model will automatically adopt a default value of 3.33 corresponding to Francis' weir formula. This value is not necessarily a good approximation.

B2.6 Other Parameters

Manning's n should be in the English system regardless of the system adopted to specify other parameters. In this model, the value is constant over the time-space domain. The appropriate value can be estimated from standard textbooks or handbooks such as Chow (1959, 1964), Henderson (1966), Brater and King (1976), and Merritt (1976). For deep reservoirs, values between 0.02 and 0.04 should be representative.

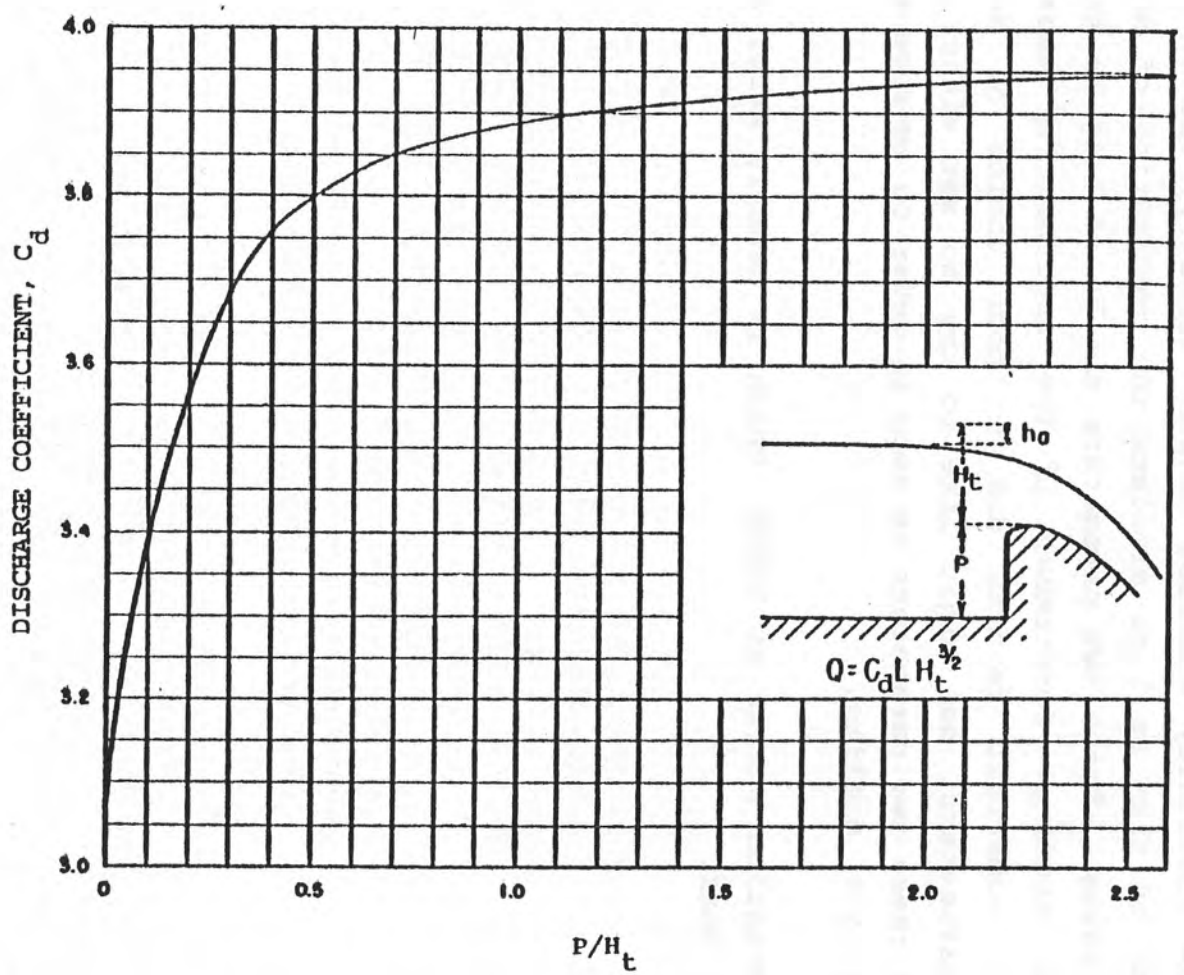


FIGURE B8: Discharge coefficient for vertical-faced ogee crest. (After U.S. Department of the Interior, 1973).

LSWAVE solves the full, nonlinear long wave equations. However, there are controlling parameters IC and ICC to change the characteristics of the mathematical model. Setting IC to zero will turn off the nonlinear convective terms in the momentum equations. Setting ICC to zero will make the continuity equation linear. Normally, one should set both of them at 1 to simulate the propagation of nonlinear waves. These two parameters can be of help in theoretical studies. Sometimes, if the nonlinearity causes abnormal numerical results due to poor choice of some model parameters, one might like to try to set either or both of these two parameters to zero in order to improve the computational quality.

The simulation starts at TZERO, which is usually zero, and ends at TMAX.

B3.0 "FORCED" LANDSLIDE

There are two ways of inputing landslide parameters in this numerical model: setting the key LSKEY = 2 for a "forced" landlside, and LSKEY = 3 for a "natural" landslide. For a "forced" landslide, the speed of the slide is specified through input data. For a "natural" landlside, the speed is computed based on initial height, slope, friction factor, and other parameters. The "forced" landslide is probably a better choice to test a landslide generated wave if the slide speed can be evaluated in advance. The user has more control on the propagation of the slide and its shape by using a "forced" landslide.

For both the "forced" and "natural" landslides, the landslide is defined by a starting point, the forward direction, the width, and the distribution of slide height (thickness) along the total length of the slide. The slide will move forward with the angle THETA in degrees measured counterclockwise from the x-axis (Figure B9). The direction will not be changed during the motion. The bathymetry and fluid pressure do not affect the course of the landslide.

The starting position is defined by the coordinate indices (XO,YO) at the right leading corner of the slide, where "right" is determined as if facing in the direction of the forward motion (Figure B9). Notice that the location of the right leading corner of the slide is designated in terms of grid units and not a distance unit like feet or meters. For a "forced" landslide, this location is allowed to be outside the M x N network, although it is usually near the waterland boundary. For a "natural" landslide, since the slide is contained inside the network, XO is a number between 1 and M, and YO between 1 and N.

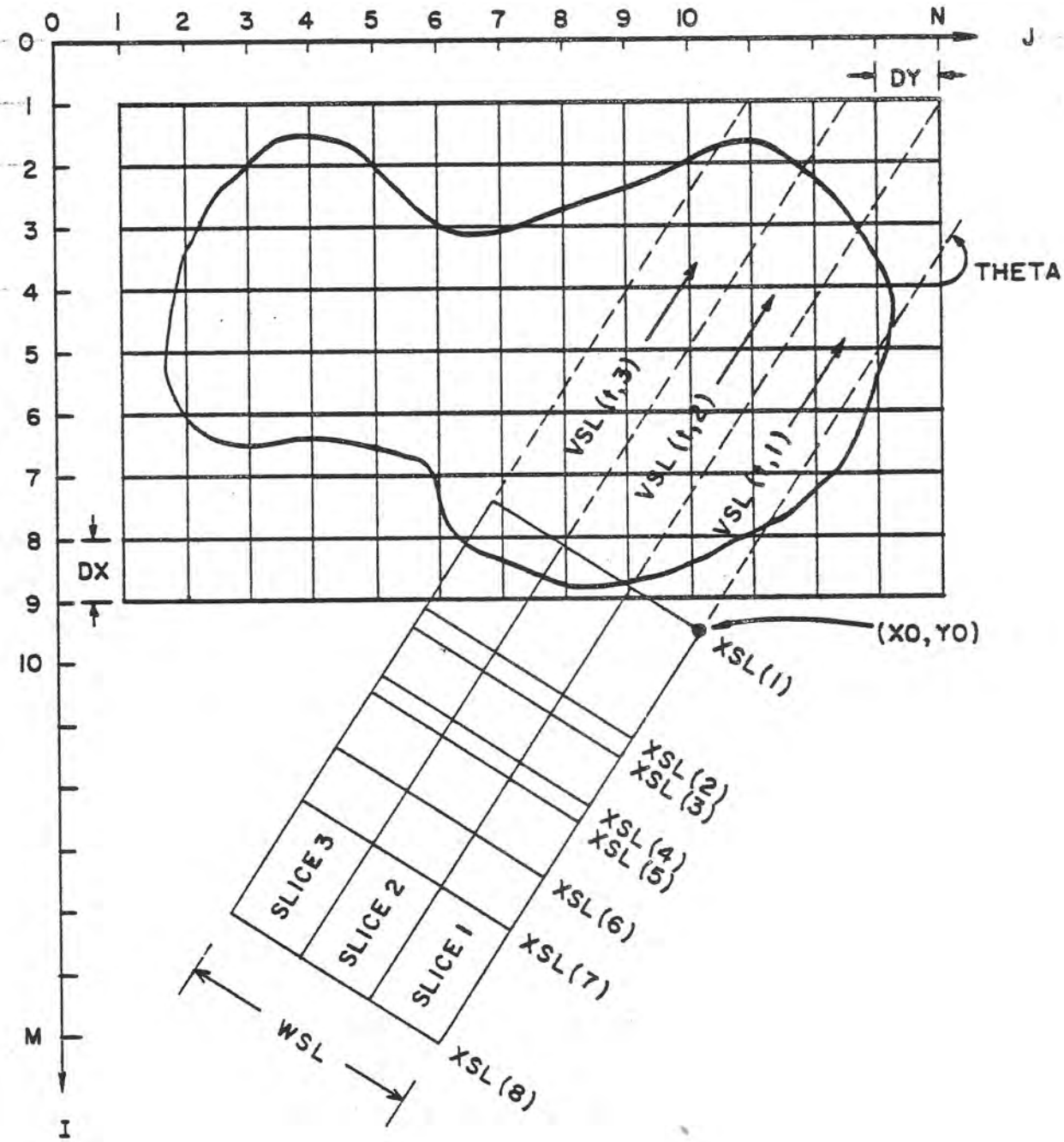


FIGURE B9: Definition sketch of landslide input data.

The landslide has a constant width, WSL. A "forced" landslide may be separated into NSL lengthwise slices each of width WSL/NSL. Each slice moves independently, i.e., they move in the same direction but with their own speed histories specified in the input data. The number of slices, NSL, is limited to 10.

Figure B9 depicts the plan view of an example "forced" landslide which intrudes into a reservoir. The computaional grid has a dimension of M x N. For this "forced" landslide M is 9. For a "natural" landslide, M should be large enough that the grid covers the landslide. The width of the landslide is WSL, which should be measured in physical length units. The number of slices, NSL, is 3 for the example "forced" landslide. A "natural" landslide can only have one slice. The forward direction of the slide, THETA, is approximately 124 degrees. The starting point, (XO,YO), is approximately (9.5, 10.1), measured in grid units. There are eight slide heights defined along the centerline of each slice, so NXH is 8. The distance from the starting point, (XO,YO), to each successive height is (XSL(I), I=1,NXH). This distance is positive from the leading edge to the trailing edge of the landslide and XSL(1) is always zero. Each slice moves independently along the dashed line until either its velocity is no longer positive, or part of the slide goes beyond the grid boundary.

The distribution of slide height (thickness) is supplied by a distance-height table. One picks up NXH nodal points along the slide, starting from the leading edge. This number, NXH, should be between 2 and 20, inclusive. The array (XSL(I), I=1,NXH) contains the horizontal distances of all nodal points, measured in a distance unit like feet or meters, from the leading edge. The interval between nodal

points is not necessarily constant but has the same spacing for each slice. The matrix ((HSL(I,J), I=1,NXH), J=1,NSL) contains the height for every point along every slice. For a "natural" landslide, this matrix is reduced to (HSL(I),I=1,NXH). The height is measured vertically, representing the difference between the top elevation and bottom elevation of the slide. The example shown in Figure B10 has 8 nodal points (NXH=8). The distance XSL(1) is always zero. The heights HSL(1) and HSL(NXH) are both zero in this example, but they may be of finite heights in certain circumstances. Heights between nodal points will be calculated by simple interpolation. An overhanging shape should be avoided.

For a "forced" landslide, one has to set another table to describe the time history of velocity for each slice. The array (TSL(I), I=1,NTV), specifies the times when the velocity is given. The variable NTV is the number of velocities specified. The time interval is not necessarily a constant. The matrix ((VSL(I,J), I=1,NTV), J=1,NSL) contains the horizontal velocity at every time for every slice. The motion of a "forced" landslide will follow this time-velocity table. It is better to choose a value of NTV such that VSL(NTV) is zero. All external forces are assumed to have been considered when this table was set up, and therefore they will not affect the pre-determined velocity during the simultation. The landslide stops when the time reaches TSL(NTV), or when the leading edge of the slide goes beyond the grid network.

When the shape of the landslide changes gradually, the grid spacings used in the hydrodynamic computations may be fine enough for this model to evaluate the change of intruding volume for every grid. However, when there is a sudden

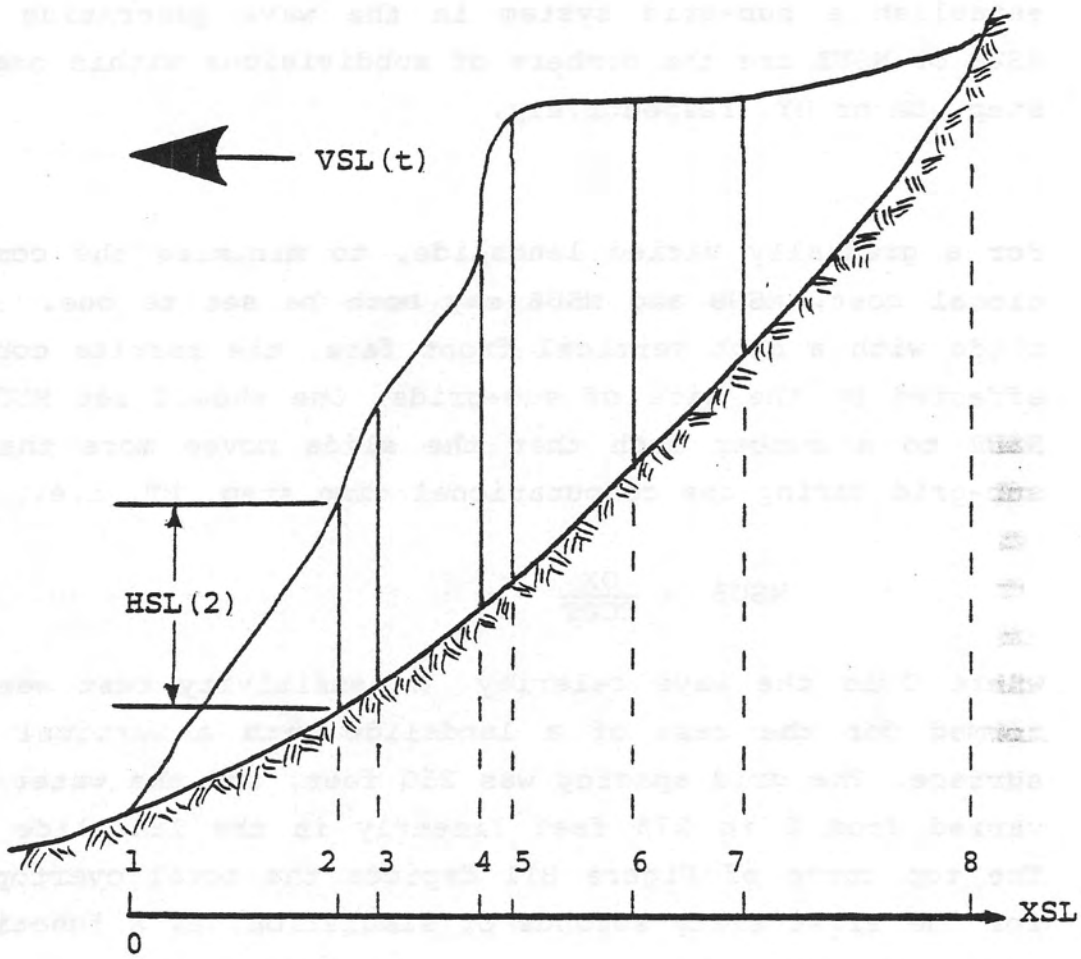


FIGURE B10: Definition sketch of landslide.

change of landslide height, HSL, within a distance of about one grid spacing, it is then better to have a finer mesh to estimate the intrusion rate. Since it is not necessary to have such a small grid spacing for the entire computational area, LSWAVE allows the user to input MSUB and NSUB to establish a sub-grid system in the wave generating area. MSUB or NSUB are the numbers of subdivisions within one grid step, DX or DY, respectively.

For a gradually varied landslide, to minimize the computational cost, MSUB and NSUB may both be set to one. For a slide with a high vertical front face, the results could be affected by the size of sub-grids. One should set MSUB and NSUB to a number such that the slide moves more than one sub-grid during one computational time step, DT, i.e.,

$$\text{MSUB} > \frac{\text{DX}}{\text{CCDT}} \tag{B6}$$

where C is the wave celerity. A sensitivity test was performed for the case of a landslide with a vertical front surface. The grid spacing was 250 feet, and the water depth varied from 0 to 275 feet linearly in the landslide area. The top curve of Figure B11 depicts the total overtopping, for the first sixty seconds of simulation, as a function of MSUB and NSUB. The curve on the bottom shows the peak discharge over the dam. The middle curve indicates the peak wave height at the upstream side of the middle section of the dam. For this particular case, MSUB and NSUB should be at least 20. A value of 50 may be used to improve the accuracy of computations. For this value and a grid spacing of 250 feet, the spacing of the sub-grid is 5 feet. The volume associated with one sub-step, across the grid of 275 feet in depth and 250 feet in width, is 275x5x250 cubic feet. When

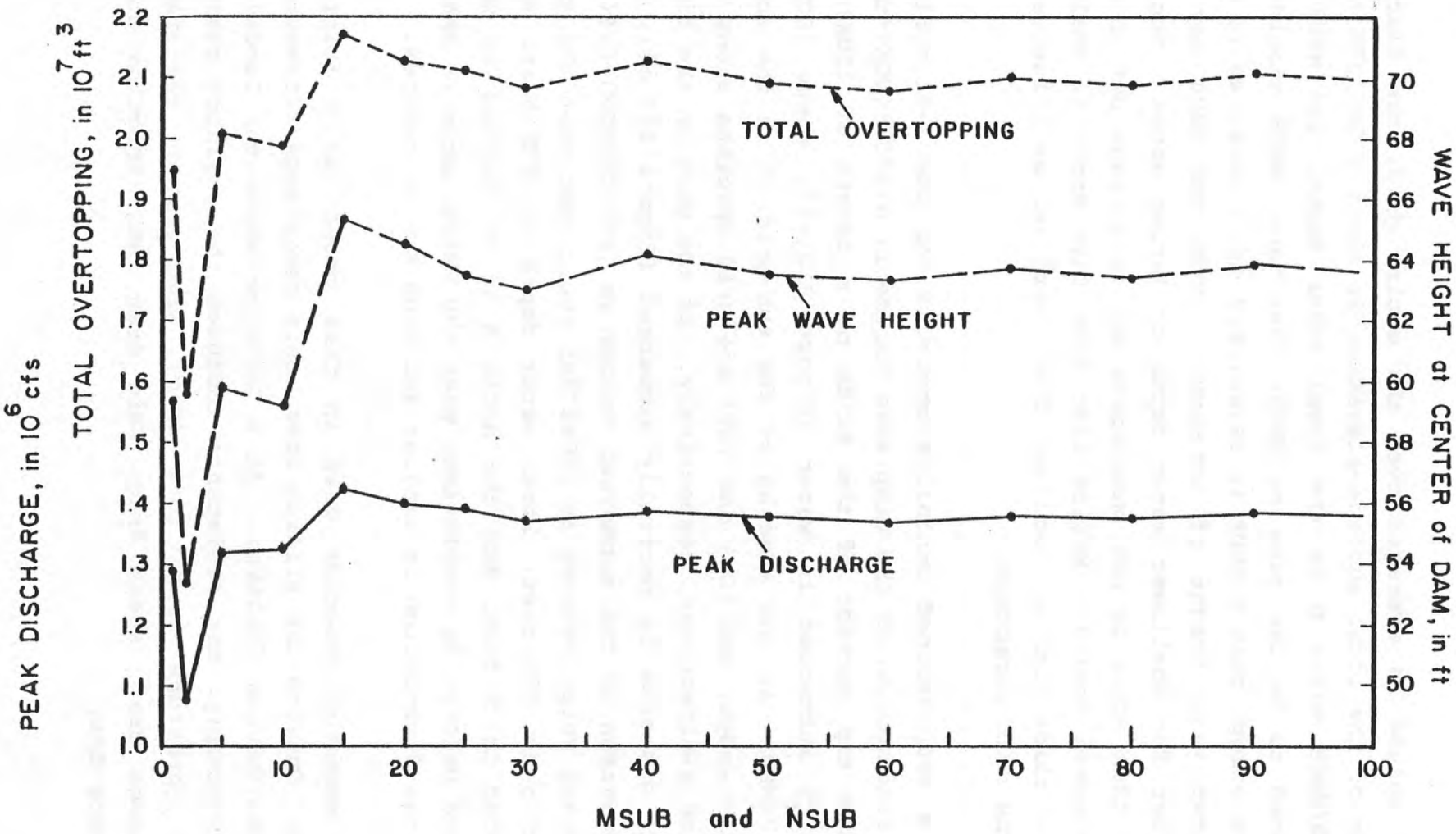
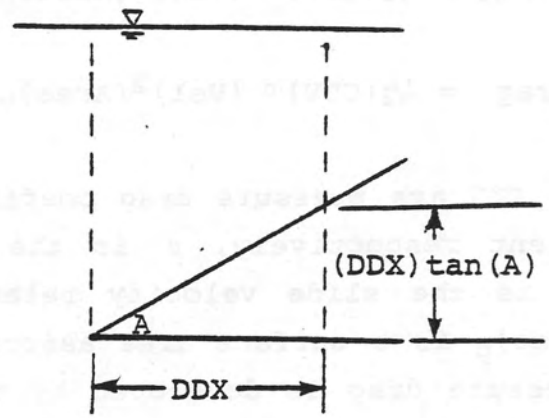


FIGURE B11: Effects of the number of sub-divisions on the peak discharge over the dam, total overtopping, and peak wave height at the center of the dam, for the case of a landslide with a vertical front surface.

this volume is averaged over the entire grid, the fluctuation of the local surface-elevation is about 275x5/250 feet or D/MSUB where D is the local water depth, and MSUB is assumed to be the same as NSUB. Therefore, MSUB should be large enough that D/MSUB is relatively small compared to the maximum wave height of interest. MSUB and NSUB may be smaller for shallower water depth or larger waves. Notice that this error is not cumulative and it spreads out in the time-space domain. Notice also that this error is smaller for a slide with an inclined front surface, as illustrated in the next paragraph.

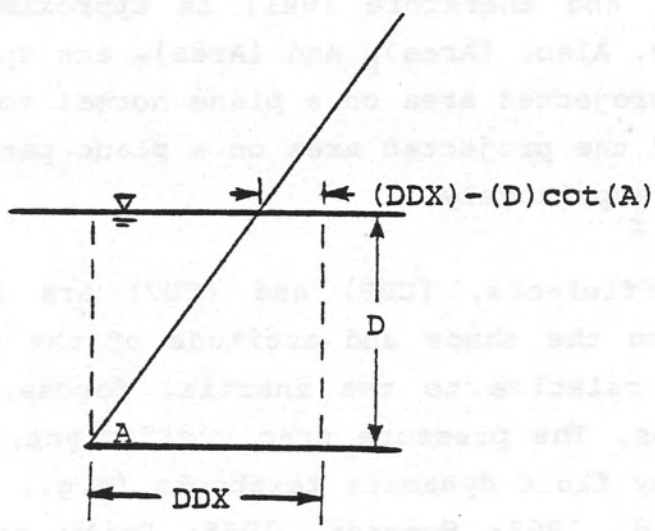
For a wedge-shaped landslide moving along the x-direction, the fluctuation of the displaced volume is $0.5(DY)(DDX)^2 \tan A$ if the tip portion of the slide of a length of (DDX) is totally submerged in water (Figure B12(a)), where (DDX)= (DX)/(MSUB) is the spacing of the sub-grid, A is the angle of the wedge, and (DX) and (DY) are grid spacings along the x- and y-directions, respectively. If the part in the first (DDX) distance is partially submerged (Figure B12(b)), the fluctuation of the submerged volume is $(DY)(D)(DDX)-0.5(DY)(D)^2 \cot A$ which reduces to 250x275x5 cubic feet when the grid spacing is 250 feet, local water depth is 275 feet, sub-spacing is 5 feet, and the angle A is 90 degrees as mentioned before. In conclusion, when the wedge angle is smaller, the fluctuation is smaller and MSUB may be reduced.

The momentum equation used in this model has a friction term. Outside the sliding area, this resistance corresponds to the bottom friction. At a location where the landslide is intruding, the resistance includes the pressure resistance (pressure drag, form drag), and viscous resistance (viscous drag, shear drag, skin drag, skin-friction drag, surface drag):



Area = $0.5(DDX)^2 \tan(A)$

(a) Submerged tip



Area = $(D)(DDX) - 0.5(D)^2 \cot(A)$

Area = $(D)(DDX)$ if A=90

(b) Partially submerged tip

FIGURE B12: Maximum error of submerged cross-sectional area for a chosen number of sub-spacings, MSUB.

$$\text{Pressure drag} = \tfrac{1}{2}(\text{CDP})\,\rho\,(\text{Vel})^2(\text{Area})_p \qquad \text{(B7)}$$

$$\text{Viscous drag} = \tfrac{1}{2}(\text{CDV})\,\rho\,(\text{Vel})^2(\text{Area})_v \qquad \text{(B8)}$$

where CDP and CDV are pressure drag coefficient and viscous drag coefficient respectively, ρ is the unit mass of the fluid, (Vel) is the slide velocity relative to the fluid velocity, $(\text{Area})_p$ is a surface area associated with viscous drag. The pressure drag is developed by the flow around an area $(\text{Area})_p$ oriented normal to the flow; the viscous drag is developed by the flow around an area $(\text{Area})_v$, parallel to the flow.

In this study, the far field velocity is taken to be negligible, and therefore (Vel) is approximated by the slide velocity. Also, $(\text{Area})_p$ and $(\text{Area})_v$ are specifically defined as the projected area on a plane normal to the slide direction, and the projected area on a plane parallel to the local slope, respectively.

Two coefficients, (CDP) and (CDV) are input data. They depend on the shape and attitude of the slide, the viscous effects relative to the inertial forces, and the boundary roughness. The pressure drag coefficient, CDP, can be found from many fluid dynamics textbooks (e.g., Schlichting, 1960, Rosenhead, 1963; Hoerner, 1965; Daily and Harleman, 1966; Batchelor, 1967; Granville, 1979). Figure B13 and Table B2 show drag coefficients which include both the pressure drag and viscous drag. These values are approximately the same as the pressure drag coefficient, since the viscous drag coefficient is relatively small. Most of the values of viscous drag coefficients in the literature are based on smooth surface objects; Figure B14 depicts the coefficient for sand-roughened plates.

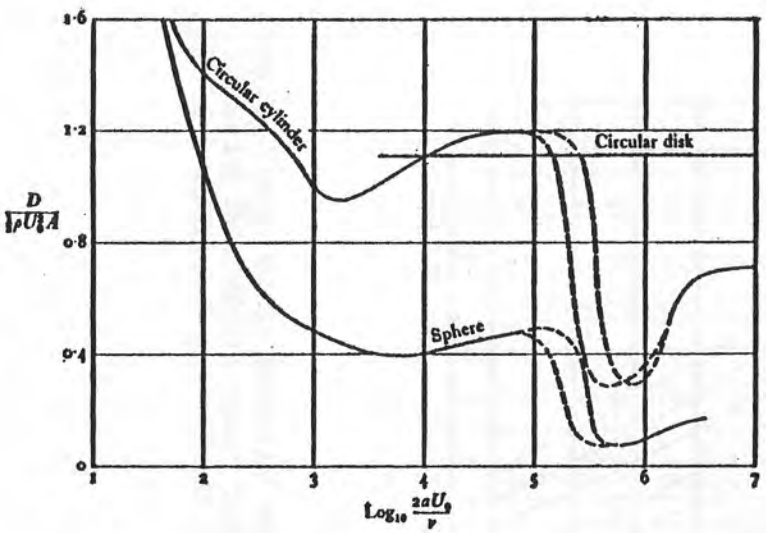


FIGURE B13: The measured drag (D consists of pressure drag and viscous drag) on unit axial length of a circular cylinder (A=2a), on a sphere ($A=\pi a^2$), and on a circular disk normal to the stream ($A=\pi a^2$), all of radius a. The broken curves represent results obtained in different wind tunnels. (After Daily and Harleman, 1966).

TABLE B2: Drag coefficients (C_D is a combination of CDP and CDV, with the area defined as the area normal to the flow). (After Daily and Harleman, 1966).

Object	c/d	L/d	$R = V_0 d/\nu$	C_D
Circular cylinder, normal to flow		1	10^5	0.63
		5	10^5	0.74
		20	10^5	0.90
		∞	10^5	1.20
		5	$>5 \times 10^5$	0.35
		∞	$>5 \times 10^5$	0.33
Elliptic cylinder, normal to flow	2	∞	4×10^4	0.60
	2	∞	10^5	0.46
	4	∞	2.5×10^4 up to 10^5	0.32
	8	∞	2.5×10^4	0.29
	8	∞	2×10^5	0.20
Square cylinder, □ normal to flow ◇		∞	3.5×10^4	2.0
		∞	10^4 up to 10^5	1.6
Circular disk, normal to flow		0	$>10^3$	1.12
Circular cylinder, parallel to flow		0	$>10^3$	1.12
		1		0.91
		4		0.87
		7		0.99
Rectangular flat plate, normal to flow		1	$>10^3$	1.10
		5		1.20
		20		1.50
		∞		2.00

L = length

d = maximum width of object measured normal to flow direction (equals minor axis of ellipsoid)

c = length of major axis of ellipsoid

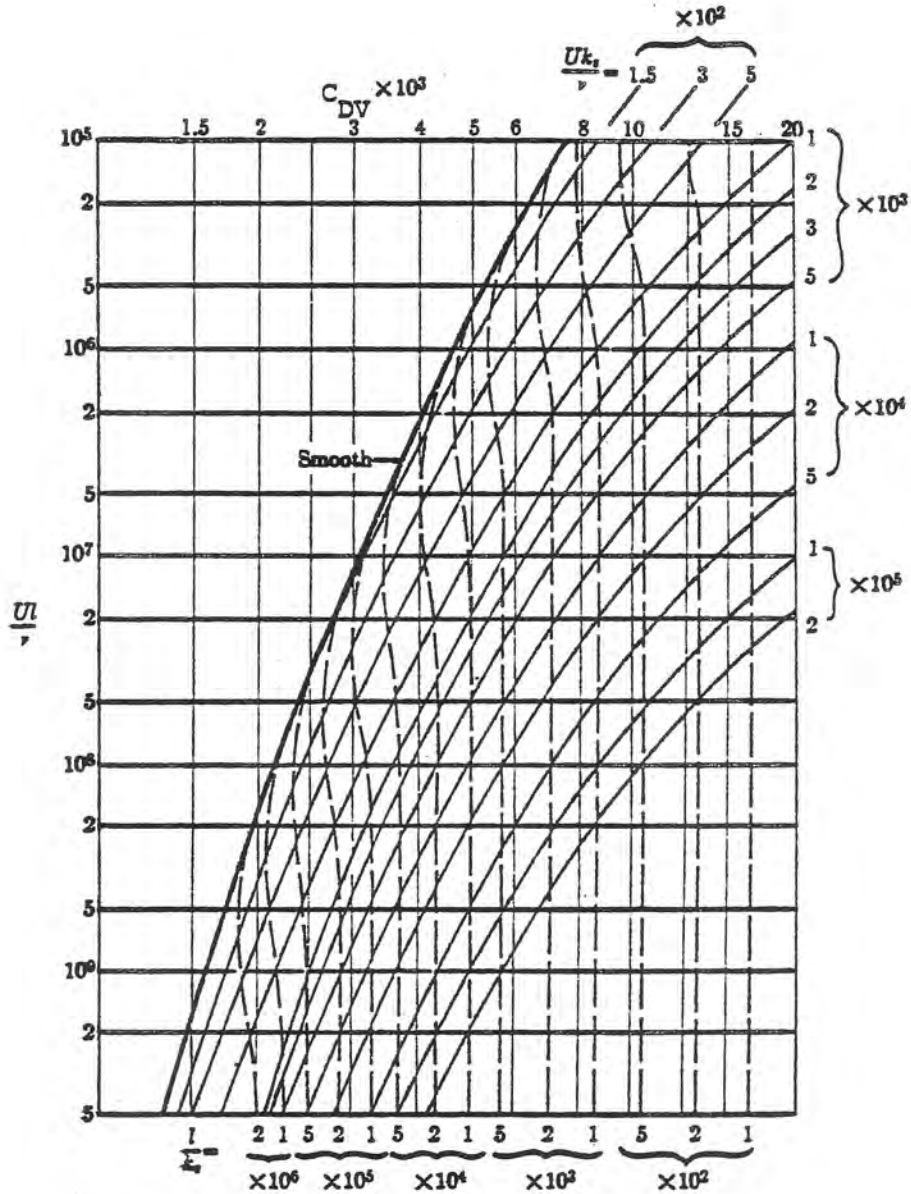


FIGURE B14: Average viscous drag coefficient, CDV, for sand-roughened plates. (After Daily and Harleman, 1966).

All calibrated values shown in the literature are steady state results. They are not necessarily good approximations for an unsteady flow. LSWAVE assumes a quasi-steady state at any time step, and applies the steady-state coefficient to obtain the drag, depending on the cross-sectional area and slide velocity.

Default values are provided for these coefficients. If the input value of either CDP or CDV is a negative number, the model will adopt the defalut value of 0.8 for CDP, or 0.008 for CDV, which are not necessarily good estimates of the coefficients under all circumstances.

B4.0 "NATURAL" LANDSLIDE

If the key LSKEY is set to 3, this model prepares the time history of the landslide velocity based on the size, shape, orientation, and starting point of the landslide, the topography, and other parameters such as friction angle, drag coefficient, and added-mass coefficient.

The input parameters are essentially the same as those for a "forced" landslide. The differences are:

(a) The user does not input the time history of the landslide velocity, i.e., TSL(I) and VSL(I,J);

(b) There is no input for NSL because LSWAVE does not allow multiple slices for the "natural" landslide;

(c) Since there are no multiple slices, the distribution of the slide height is described by a one-dimensional array HSL(I), instead of HSL(I,J);

(d) It requires additional input parameters, SGLS, PHI, and AK, which are discussed in the following paragraphs.

The symbol SGLS denotes the specific gravity of the whole slide (not that of the soil particle) before the landslide occurs. It can be obtained from

$$(SGLS) = (W_{slide}) / (W_{water}) \qquad (B9)$$

where W can be the unit mass, the unit weight (density), or the weight of any specific volume (see for example, Wu, 1966). If the input for this variable is a negative value, it is defaulted to 1.7 which is equivalent to a dry density of 106 pcf. The specific gravity of the slide composed of dry soil is about 1.7 or 1.8; while that of rock is much higher as can be seen from Table B3.

The symbol PHI denotes the angle of friction (angle of shearing resistance), in degrees, between the slide and the

TABLE B3

SPECIFIC GRAVITY AND UNIT WEIGHT OF VARIOUS ROCKS.
(After Krynine and Judd, 1957)

Rock type	No. of specimens	Apparent sp. gr.	Porosity, %	Sorption, %	Dry unit wt, pcf
Igneous:					
Andesite	4	2.22	10.77	4.86	174.3
		2.70	0.72	0.28	168.5
		2.79	0.10	0.05	138.8
Aplite	1	*2.50*†	*4.11*	*1.67*	*156.1*
Basalt	4	2.77	22.06	9.97	172.9
		2.75	1.10	0.13	171.6
		2.21	0.22	0.66	138.2
Dacite	1	*2.46*	*3.50*	*1.44*	*153.6*
Diabase	2	2.95	0.17	0.06	184.4
		2.82	1.00	0.38	176.0
Gabbro	3	3.00	0.00	0.00	187.3
		2.86	0.29	0.13	178.4
		2.72	0.62	0.25	169.6
Granite	17	2.67	3.98	1.55	166.6
		2.60	1.11	0.44	162.7
		2.53	0.44	0.20	158.2
Granite (fluorite)	1	*2.99*	*1.67*	*0.58*	*186.6*
Granodiorite	1	*2.70*	*0.50*	*0.19*	*168.5*
Quartz Syenite	1	*2.63*	*1.54*	*0.62*	*164.1*
Rhyolite	1	*2.49*	*4.13*	*1.67*	*155.2*
Sedimentary:					
Breccia (syenite)	1	*3.10*	*0.78*	*0.27*	*193.3*
Breccia (limestone)	1	*2.28*	*18.73*	*8.26*	*142.1*
Chert	1	*2.48*	*4.10*	*1.69*	*155.1*
Coquina	1	*1.19*	*56.70*	*47.80*	*74.0*
Coral	2	2.66	1.06	0.41	166.2
Limestone	7	*2.54*	*4.36*	*1.73*	*158.9*
		2.67	1.70	0.65	166.8
		2.72	*0.27*	*0.12*	*169.7*
Limestone (dolomitic)	2	2.69	2.08	0.80	167.9
Limestone (oolitic)	3	2.66	1.06	0.42	166.3
Sandstone	6	*2.58*	*1.62*	*0.66*	*160.9*
		2.35	9.25	4.12	147.2
		1.91	*26.40*	*13.80*	*119.5*
Sandstone (calcareous)	1	*2.31*	*11.85*	*5.14*	*144.5*
Sandstone (clayey)	1	*2.48*	*6.10*	*2.43*	*155.1*
Travertine (onyx)	2	2.63	1.07	0.42	164.7
Metamorphic:					
Gneiss	5	3.12	2.23	0.84	195.0
		2.66	0.78	0.30	166.1
		2.61	0.30	0.12	162.9
Marble	7	2.73	2.02	0.77	170.2
		2.61	0.62	0.23	163.1
		2.49	0.31	0.13	155.5
Marble (dolomitic)	2	2.84	0.60	0.21	177.2
Quartzite	3	2.64	0.46	0.17	164.7
Slate	3	*2.77*	*0.00*	*0.00*	*173.0*
		2.74	1.06	0.40	171.5

* From J. H. Griffith, Physical Properties of Typical American Rocks, *Iowa Eng. Expt. Sta. Bull.* 131, March, 1937.

† Values in italics are the results of tests on only one rock specimen.

Where three values are given for one rock type, the middle value is the average for that particular physical property. However, the highest and lowest values shown were not used in computing the average value.

sliding slope. The angle of internal friction for a silt or uniform fine to medium sand is 26-30 degrees, that for a well-graded sand is 30-34 degrees, and that for sand and gravel is 32-36 degrees (Merritt, 1976, p. 7-9). The friction angle for clay is about 0-10 degrees. If the input value is negative, it is defaulted to 10 degrees. The friction includes the effects of the slide riding on an air cushion, if any, and the negligible adhesion between clay and sliding surface.

The symbol AK stands for the added-mass coefficient. The product of this coefficient and the mass of water which has a volume equal to the submerged slide is the added water mass to be accelerated by the external forces. This coefficient depends on the shape and attitude of the slide. It is defaulted to 0.2 if input with a negative value. The value shown in the literature (Stelson and Maris, 1957; Daily and Harleman, 1966) is generally higher than 0.2. For example, it is 1.0 for a long cylinder (smaller if the length-diameter ratio is smaller), 0.67 for a cube acclerated either "broadside-on" or "edge-on", 0.50 for a sphere, and 0.20 for an ellipsoid accelerating "end-on" with the major axis twice the minor axis. A low value of 0.2 was selected as the default value because this case is a landslide in a reservoir of free-surface, where the water is allowed to have vertical acceleration.

The "natural" landslide acquires an acceleration caused by effects of the gravitational force, friction between the slide and the sliding surface, form drag, viscous drag, added mass, and buoyancy. The slide comes to a stop either when the velocity is no longer positive, or when a part of the slide goes beyond the M x N grid network.

B5.0 BOTTOM MOTION GENERATED WAVES

When the code LSKEY is set to 4, this model simulates the propagation of waves generated by bottom motion. The bottom motion is described by a rise (or subsidence) of the area to one side of a fault line. The fault line is defined by two points which are not necessary inside the grid network. These two points have coordinates (FI(1),FJ(1)) and (FI(2),FJ(2)), respectively. The area to the right of the fault line, when standing at the first point and facing the second, will have an instantaneous rise of magnitude DFLT (negative for subsidence). The wave is generated by the instantaneous rise or subsidence of the water surface in that area (shaded area in Figure B15). Note that both the surface elevations and the water depths are adjusted at the initial instant.

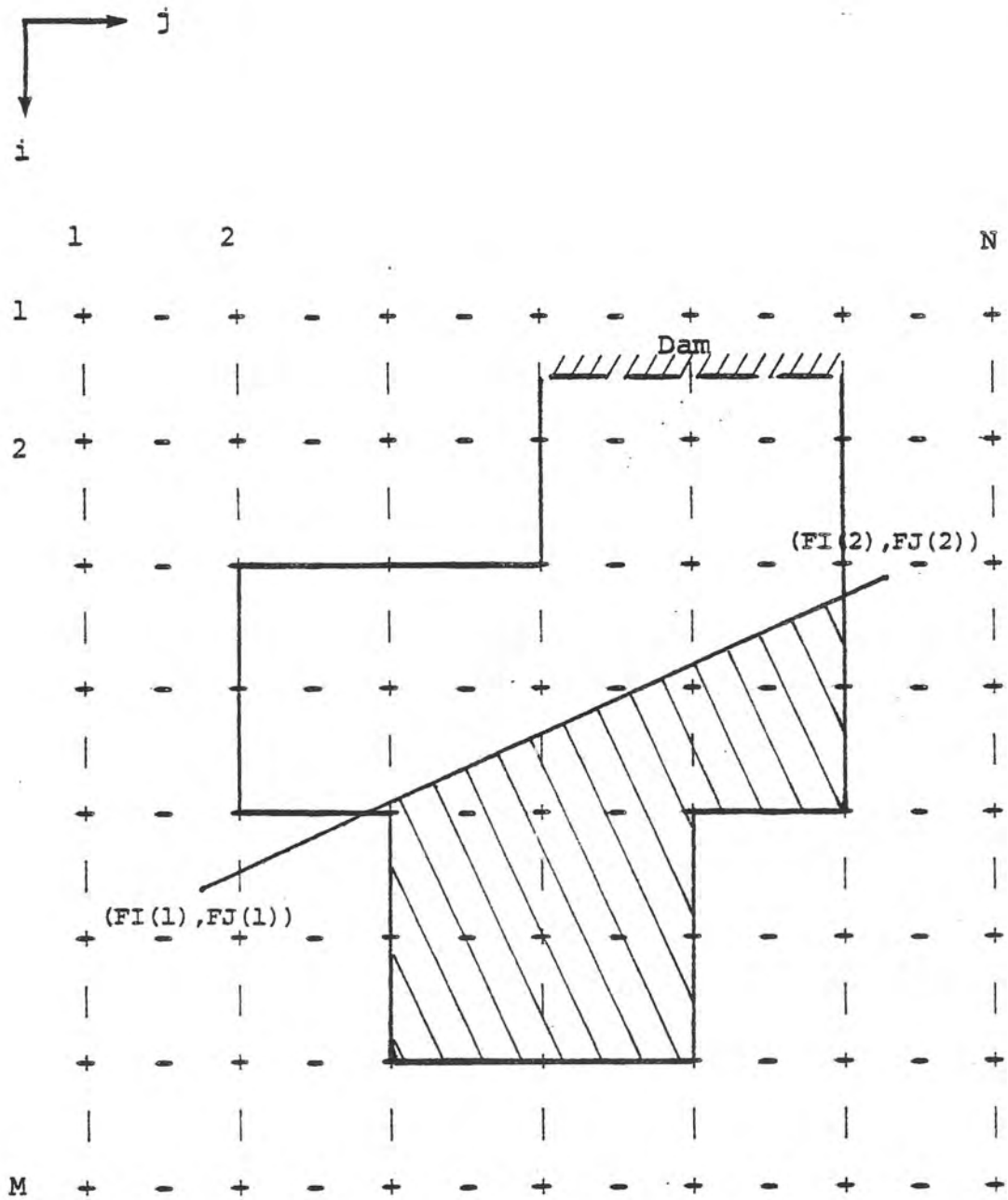


FIGURE B15: Definition sketch of fault-induced bottom motion.

B6.0 DESCRIPTION OF INPUT PARAMETERS

Group No. and Format	Variables	Description
1. 8A10 (1-10 Cards)	ID(K)	One to ten title cards. Columns 1 through 79 of each card contain descriptive information. Column 80, of all but the last card, must be "+". The character "+" in column 80 may be thought of as a continuation character. One card is necessary, and it can be blank. If a single card is used, column 80 must not contain "+". For each card, the information, from column 1 to the last punched character, except "+" in column 80, will be centered on printed output.
2. 8I5 (1 Card)	M	Dimension of grid in x-direction.
	N	Dimension of grid in y-direction (M*N.LE.2000).
	MOB	Number of open boundary segments normal to the x-axis (MOB.LE.4).
	NOB	Number of open boundary segments normal to the y-axis (NOB.LE.4).
	MWEIR	Number of dam segments normal to the x-axis (MWEIR.LE.9).
	NWEIR	Number of dam segments normal to the y-axis (NWEIR.LE.9).
	IC	Controls exclusion of convective terms in momentum equation (0 = excluded; 1 =included).
	ICC	Controls exclusion of non-linear term in continuity equation (0 = excluded; 1 = included).
3. 7F10.5 (1 Card)	TZERO	Time to be considered the starting time for the simulation (seconds). (Usually 0).

	TMAX	Ending time (seconds). Total simulation time is equal to TMAX-TZERO.
	DT	Computational time step (seconds). It has to satisfy the Courant stability condition such that it does not exceed DX*DY/ (SQRT((DX*DX+DY*DY)*G*Dmax) * (1+Emax/(Dmax+Emax))) where Dmax is the maximum water depth relative to the initial water surface and Emax is the maximum surface elevation above the initial level.
	DX	X-direction distance between grid points in desired units. (Be sure that units are consistent with all other input quantities involving distance.)
	DY	Y-direction grid spacing.
	DCONV	Conversion factor for the input grid values. Depth = DCONV *(Z0-D(K)) if Z0.NE.0. and D.NE.0. Depth = DCONV * D(K) if Z0=0.
	Z0	Initial water surface elevation. If depths are input for the grid, rather than elevations, then set Z0=0.
4. 4F10.5 (1 Card)	CMANN	Manning roughness coefficient. (Use the value in FPS unit system no matter which unit system is used for other parameters.)
	CED	Energy dissipation coefficient for partially reflective boundaries.
	CD	Discharge coefficient for the weir formula, with a dimension of SQRT(L)/(T). It becomes the default value of 3.33 if the input value is .LE.0. One should not use this default value if the unit system is not FPS.
	G	Gravitational acceleration (in consistent units).

5. 14I5 (1 card)	MES	Number of grid points for which histories will be stored (MES.LE.50).
	IS	Number of time steps between stored history values. If MES is large and IS is small, it is possible that the storage alotted for histories will be exceeded. If this occurs, the program will give an error message and stop. User may decrease MES, decrease TMAX, or increase IS to recover.Doubling IS will save half of the printing space but precision will be lost in the plotting and in the printed results. IS = 1 will yield the most information.
	ISP	This value times IS yields the number of steps between printed history values. ISP = 1 will print the most information.
	ILR(I,J)	Controls plotting and printing of histories of surface elevation, velocity and over-topping. J = 1: surface elevation J = 2: velocity J = 3: over-topping I = 1: if 0, histories not plotted if 1, plot, half page each if 2, plot, full page I = 2: if 0, histories not printed (Order of input is with I varying most rapidly.)
	KS	Number of snapshots of surface elevation and/or velocity. The times for these snapshots are specified in TS(K) in Group 13 (KS.LE.21).
	LSE	Controls printing of snapshots of surface elevation (0 = do not print; 1 = print).
	LSU	Controls printing of snapshots of velocity. (0 = do not print; 1 = print).
	IENV	Controls printing of maximum value of surface elevation (0 = do not print; 1 = print).

	IPLTF	Controls writing of histories to an unformatted file (Tape 2) for external plotting. (If .NE.0 file is written).
6. 2F10.5	EMAX	Maximum range for plots of surface elevation histories (range will be from -EMAX to +EMAX). If EMAX = 0, each plot will be scaled by the program.
	UMAX	Maximum range for velocity plots, similar to EMAX.
7. 16F5.0 (M*(INT((N-1)/16+1) Cards)	D(K)	Array of depths or elevations. Y-direction is across a data card; each row should start a new card (e.g., if M=5 and N=7, there should be 5 cards, each with 7 values; if M=5 and N=17, there should be 10 cards, the first with 16 values, the second with 1, the third with 16, etc.). If the input value is equal to zero (or left blank), the point is assumed to be outside the computational region; this is true regardless of the value of Z0 in Group 3. If depths are input (Z0=0) then downward is positive, measured from still surface elevation. If elevations are input (Z0.NE.0.), then upward is positive, measured from the same origin as Z0. The following rules are followed in converting from input values to depths: Depth = DCONV*(Z0-D(K)) if Z0.NE.0. and D.NE.0. Depth = DCONV*D(K) if Z0=0.

****

In Groups 8-13, use no cards in any group for which the control variable (MOB, NOB, MWEIR, NWEIR, MES, OR KS) is equal to zero.

****

8. 4(3I3,1X) (1 Card or no card)	(IOB(K), J1OB(K), J2OB(K), K=1,MOB)	Locations of open boundary segments normal to the x-axis: IOB is the row in which a segment lies and J1OB, J2OB are the columns at the end points of the segment (J1OB.LE.J2OB).

9. 4(3I3,1X) (1 Card or no Card)	(JOB(K), I1OB(K), I2OB(K), K=1,NOB)	Locations of open boundary segments normal to the y-axis: JOB is the column in which a segment lies and I1OB, I2OB are the rows at the end points of the segment (I1OB.LE.I2OB).
10. 4(3I3,1X,F10.5) (0-3 Cards)	(IWEIR(K), J1WEIR(K), J2WEIR(K), HWEIRI(K),	Locations and heights of dam segments normal to the x-axis. Locations are specified in the same way as for open boundaries. HWEIRI is the height above still water surface (freeboard of the dam). Each dam which contains one or more segments is allowed only one HWEIRI value and one HWEIRJ value (Normally, HWEIRI and HWEIRJ are the same for a single dam.)
11. 4(3I3,1X,F10.5) (0-3 Cards)	(JWEIR(K) I1WEIR(K), I2WEIR(K), HWEIRJ(K), K=1,NWEIR)	Locations and heights of dam segments normal to the y-axis.
12. 8(2I5) (INT((MES-1) /8+1 Cards)	(IHIST(K), JHIST(K), K=1,MES)	Locations for which histories are to be stored. IHIST is the row and JHIST is the column.
13. 8F10.5 (INT((KS-1) /8+1) Cards)	(TS(K), K=1,KS)	Times (seconds) at which snapshots are desired.
14. I1,9X,7A10 (1 Card)	LSKEY	1 for no landslide (no use at present); 2 for "forced" landslide; 3 for "natural" landslide; 4 for bottom movement due to fault; 5 for user defined subroutines.
	TITLE(K)	Sub-title for the landslide (0 to 70 characters).

****

If LSKEY = 1, skip the following groups;
If LSKEY = 2, skip Groups 18 and 19'
If LSKEY = 3, skip Groups 17, 19, 22, and 23;
If LSKEY = 4, skip Groups 15-18 and 20-23;
If LSKEY = 5, skip the following groups.

****

15. 5I5 (Include this card if LSKEY = 2 or 3)
(1 Card)

NSL — Number of slices in "forced" landslide (NSL.LE.10); Default to 1 if LSKEY = 3.

NXH — Number of heights specified for each slice (NXH.LE.100).

NTV — Number of velocities specified for each slice (NTV.LE.100). Not used if LSKEY = 3.

MSUB — Number of sub-spacings for a DX, in the landslide area; default to 10 if the input value is zero or negative (MSUB.LE.50).

NSUB — Number of sub-spacings for a DY; default to 10 if the input value is zero or negative (NSUB.LE.50).

16. 6F10.5 (Include this card if LSKEY = 2 or 3).
(1 Card)

THETA — Forward direction of slide (degrees counterclockwise from x-axis).

WSL — Width of slide. The width of a slice for a "forced" landslide is WSL/NSL.

XO — X-coordinate (grid units, not distance) of right leading corner of slide ("right" is determined as if facing in the direction of forward motion).

YO — Y-coordinate of right leading corner of slide (the coordinates are not necessary round numbers).

CDP — Pressure drag (form drag) coefficient; defaults to 0.8 if input with a negative number.

	CDV	Viscous drag (skin drag) coefficient; defaults to 0.008 if input with a negative number.
17. 2F10.5 (1 Card)	(Include this card if LSKEY = 2).	
	TSLCNV	Factor to convert TSL(I) in Group 22 to seconds.
	VSLCNV	Factor to convert VSL(I,J) in Group 23 to appropriate units.
18. 3F10.5 (1 Card)	(Include this card if LSKEY = 3).	
	SGSL	Specific gravity of the whole landslide; default to 1.7 if input with a negative number.
	AK	Added mass coefficient; default to 0.2 if input with a negative number.
	PHI	Friction angle in degrees, between slide and sliding slope; default to 10° if input with a negative number.
19. 5F10.5 (1 Card)	(Include this card if LSKEY = 4).	
	(FI(K) FJ(K), K=1,2)	Coordinates, not necessarily integers, of two points on fault line; displaced region is to the right of the line ("right" is determined as if standing at the first point, facing the second). The order is FI(1), FJ(1), FI(2), and FJ(2).
	DFLT	Magnitude of bottom displacement to the right of the fault line specified by two points, (FI(1), FJ(1)) and (FI(2), FJ(2));positive for upward motion.
20. 8F10.5 (INT((NXH-1) /8+1)Cards)	(Include this Group of cards if LSKEY=2 or 3).	
	(XSL(I), I=1,NXH)	Locations where slide height is specified (distances from leading edge, in distance units).
21. 8F10.5 (NSL sets of (INT((NXH-1 /8+1)Cards)	(Include this Group of cards if LSKEY=2 or 3).	
	(HSL(I,J), I=1,NXH) for J=1,NSL	Height of slice J at distance XSL(I) from leading edge.

22. 8F10.5 (INT((NTV-1) /8+1)Cards)	(TSL(I), I=1,NTV)	(Include this Group of cards if LSKEY = 2). Times at which slide velocity is specified.
23. 8F10.5 (NSL sets of (INT((NTV-1) /8+1)Cards)	(VSL(I,J), I=1,NTV), for J=1,NSL	(Include this Group of cards if LSKEY = 2). Horizontal velocity of slice J at time TSL(I).

TABLE B4

INDEX OF INPUT VARIABLES

Variable	Group	Variable	Group
AK	18	JIWEIR(K)	10
CD	4	J2OB(K)	8
CDP	16	J2WEIR(K)	10
CDV	16	KS	5
CMANN	4	LSE	5
CT	4	LSKEY	14
D(K)	7	LSU	5
DCONV	3	M	2
DFLT	19	MES	5
DT	3	MOB	2
DX	3	MSUB	15
DY	3	MWEIR	2
EMAX	6	N	2
FI(K)	19	NOB	2
FJ(K)	19	NSL	15
G	4	NSUB	15
HSL(I,J)	21	NTV	15
IC	1	NWEIR	2
ICC	1	NXH	15
ID	1	PHI	18
IENV	5	SGSL	18
IHIST(K)	12	THETA	16
ILR(I,J)	5	TITLE(K)	14
IOB(K)	8	TMAX	3
IPLTF	5	TS(K)	13
IS	5	TSL(I)	22
ISP	5	TSLCNV	17
IWEIR(K)	10	TZERO	3
I1OB(K)	9	UMAX	6
I1WEIR(K)	11	VSL(I,J)	23
I2OB(K)	9	VSLCNV	17
I2WEIR(K)	11	WXL	16
JHIST(K)	12	XSL(I)	20
JOB(K)	9	XO	16
JWEIR(K)	11	YO	16
J1OB(K)	8	ZO	3

B7.0 OUTLINE OF OUTPUT DATA

The computer output consists of:

(a) a list of input parameters;

(b) some auxiliary information regarding input data;

(c) optional "snapshots" of surface elevation and/or velocity patterns;

(d) optional listing of time histories of surface elevations, velocities, and/or discharge over the dam;

(e) optional printer plotted time histories of surface elevations, velocities, and/or discharge over the dam;

(f) data file TAPE1 which contains the time history of the discharge over the dam, to be used as input data for the downstream flood routing model;

(g) optional data file TAPE2 which contains the time histories of the surface elevations and/or velocities at specified grid points and the time history of discharges over the dam, to be used as input data for pen plotter routines.

TAPE1 stores (TZERO, DTIS, NES, (Q(I), I=1,NES)) without format, where DTIS, the time interval between the stored discharges, is the product of DT and IS. The array Q contains the time history of discharge over the dam.

Data (TZERO, DTIS, NES, MES, (IHIST(I), JHIST(I), I=1,MES); (ES(I), I=1,NMES); (US(I), I=1,NMES); (VS(I), I=1,NMES); and (Q(I), I=1,NMES)) are stored in TAPE2 through five unformatted write statements. DTIS is the product of DT and IS. To read data from TAPE2, NMES should be computed after NES and MES are read and before ES(I) are read. The value of NMES is the product of NES and MES.

To obtain the wave data before an earth dam, one may choose one or more representative locations before the dam, so that the time histories of the surface fluctuations will be printed in the output and/or stored in the file TAPE2.

B8.0 PROGRAM LISTING

```
      PROGRAM LSWAVE (INPUT,OUTPUT,TAPE1,TAPE2)                         LSWAVE.2
C                                                                       LSWAVE.3
C     VERSION A.3, NOVEMBER 15, 1981                                    LSWAVE.4
C     DEVELOPED BY TETRA TECH, INC., PASADENA, CALIFORNIA               LSWAVE.5
C                                                                       LSWAVE.6
C ***********************************************************************LSWAVE.7
C **  LIST OF VARIABLES IN MAIN PROGRAM AND SUBPROGRAMS INITIAL,      **LSWAVE.8
C **    SIMULAT, FEEDBKU, FEEDBKV, DATARDR, DATAPRT                   **LSWAVE.9
C ***********************************************************************LSWAVE.10
C                                                                       LSWAVE.11
C     C(*)          = CC*H(*)**(4./3.)                                  LSWAVE.12
C     CC            = (1.486/CMANN)**2/(G*DT)/2.                        LSWAVE.13
C     CD            DISCHARGE COEFFICIENT FOR THE WEIR FORMULA          LSWAVE.14
C     CD2           = CD*2.                                             LSWAVE.15
C     CDTDX         = DTDX*IC                                           LSWAVE.16
C     CDTDX5        = CDTDX*0.5                                         LSWAVE.17
C     CDTDY         = DTDY*IC                                           LSWAVE.18
C     CDTDY5        = CDTDY*0.5                                         LSWAVE.19
C     CMANN         MANNING ROUGHNESS COEFFICIENT                       LSWAVE.20
C     CED           COEFFICIENT OF ENERGY DISSIPATION AT PARTIALLY      LSWAVE.21
C                     REFLECTIVE BOUNDARIES                             LSWAVE.22
C     CTMR          = CED-(1.-CED)                                      LSWAVE.23
C     C3X           = SQRT(G*0.5)*DTDX                                  LSWAVE.24
C     C3Y           = SQRT(G*0.5)*DTDY                                  LSWAVE.25
C     D(*)          DEPTH                                               LSWAVE.26
C     DCONV         CONVERSION FACTOR FOR INPUT DEPTHS                  LSWAVE.27
C     DT            COMPUTATIONAL TIME STEP (SECONDS)                   LSWAVE.28
C     DTDX          = DT/DX                                             LSWAVE.29
C     DTDX5         = DTDX*0.5                                          LSWAVE.30
C     DTDY          = DT/DY                                             LSWAVE.31
C     DTDY5         = DTDY*0.5                                          LSWAVE.32
C     DX            GRID SPACING IN X-DIRECTION                         LSWAVE.33
C     DXDT2         = (DX+DX)/DT                                        LSWAVE.34
C     DXDT2C        = DXDT2*IC                                          LSWAVE.35
C     DX5           = DX*0.5                                            LSWAVE.36
C     DY            GRID SPACING IN Y-DIRECTION                         LSWAVE.37
C     DYDT2         = (DY+DY)/DT                                        LSWAVE.38
C     DYDT2C        = DYDT2*IC                                          LSWAVE.39
```

```
C     DY5          = DY*0.5                                                LSWAVE.40
C     D0(*)        DUMMY VECTOR                                            LSWAVE.41
C     D1(*)        DUMMY VECTOR                                            LSWAVE.42
C     E(*)         SURFACE ELEVATION                                       LSWAVE.43
C     EH(*)        MAXIMUM SURFACE ELEVATION                               LSWAVE.44
C     EMAX         IF .GT. 0., RANGE USED FOR ELEVATION PLOTS IS           LSWAVE.45
C                    -EMAX TO +EMAX                                        LSWAVE.46
C     ES(*)        HISTORIES OF SURFACE ELEVATION                          LSWAVE.47
C     G            GRAVITATIONAL ACCELERATION                              LSWAVE.48
C     GDTDX        = G*DTDX                                                LSWAVE.49
C     GDTDY        = G*DTDY                                                LSWAVE.50
C     G5           = 0.5/G                                                 LSWAVE.51
C     H(*)         = HT(*) (IF ICC=1),  = D(*) (IF ICC=0)                  LSWAVE.52
C     HIJ          = H(MAP(I,J))                                           LSWAVE.53
C     HT(*)        = E(*)+D(*)                                             LSWAVE.54
C                 HEIGHT OF DAM ABOVE STILL WATER LEVEL (FREEBOARD OF DAM)LSWAVE.55
C     HWEIRI(*)      FOR DAM SEGMENT NORMAL TO X-AXIS                      LSWAVE.56
C     HWEIRJ(*)      FOR DAM SEGMENT NORMAL TO Y-AXIS                      LSWAVE.57
C     HWEI1          FOR LOWER END OF SEGMENT NORMAL TO X-AXIS             LSWAVE.58
C     HWEI2          FOR UPPER END OF SEGMENT NORMAL TO X-AXIS             LSWAVE.59
C     HWEJ1          FOR LOWER END OF SEGMENT NORMAL TO Y-AXIS             LSWAVE.60
C     HWEJ2          FOR UPPER END OF SEGMENT NORMAL TO Y-AXIS             LSWAVE.61
C     H2U(I,J)     = (H(I,J)+H(I+1,J))*UN(I,J)                             LSWAVE.62
C     H2V(I,J)     = (H(I,J)+H(I,J+1))*VN(I,J)                             LSWAVE.63
C     I            ROW INDEX                                               LSWAVE.64
C     IC           IF 0, CONVECTIVE TERM IS EXCLUDED                       LSWAVE.65
C     ICC          IF 0, CONTINUITY EQUATION IS LINEAR                     LSWAVE.66
C     IDIMD        DIMENSION OF D,D0,D1,MAP                                LSWAVE.67
C     IDIME        DIMENSION OF E,C,H,HT,H2U,H2V,U,UN,V,VN                 LSWAVE.68
C     IDIMEH       DIMENSION OF EH                                         LSWAVE.69
C     IDIMOB       MAXIMUM NUMBER OF OPEN BOUNDARY SEGMENTS IN EITHER      LSWAVE.70
C                    DIRECTION (DIMENSION OF IOB,I1OB,I2OB,JOB,J1OB,J2OB) LSWAVE.71
C     IDIMTS       MAXIMUM NUMBER OF SNAPSHOTS (DIMENSION OF TS,ITS)       LSWAVE.72
C     IDIMWE       MAXIMUM NUMBER OF DAM SEGMENTS IN EITHER DIRECTION      LSWAVE.73
C                    (DIMENSION OF IWEIR,I1WEIR,I2WEIR,JWEIR,J1WEIR,       LSWAVE.74
C                    J2WEIR,HWEIRI,HWEIRJ)                                 LSWAVE.75
C     IENV         IF 0, MAXIMUM SURFACE ELEVATION NOT PLOTTED             LSWAVE.76
C     IES          COUNTER FOR HISTORIES                                   LSWAVE.77
```

```
C       IHIST(*)    I-COORDINATE OF POINT FOR WHICH A HISTORY IS STORED     LSWAVE.78
C       I11         DUMMY INTEGER VARIABLE                                  LSWAVE.79
C       I12         DUMMY INTEGER VARIABLE                                  LSWAVE.80
C       IJ          = MAP(I,J), RANGING FROM 2 TO MAXCPM                    LSWAVE.81
C                     ALSO, DUMMY INTEGER VARIABLE                          LSWAVE.82
C       IJM         = MAP(I,J-1)                                            LSWAVE.83
C       IJP         = MAP(I,J+1)                                            LSWAVE.84
C       IK(*)       ROW NO. OF COMPUTATIONAL SEGMENT NORMAL TO X-AXIS       LSWAVE.85
C       IL          DUMMY INTEGER VARIABLE                                  LSWAVE.86
C       ILR(*)      CONTROLS PLOTTING AND PRINTING OF HISTORIES             LSWAVE.87
C                     ILR(I,J) = 0, NO PLOTTING OR PRINTING                 LSWAVE.88
C                     ILR(1,J) = 1, PLOTTING (VERTICAL AXIS HALF PAGE)      LSWAVE.89
C                     ILR(1,J) = 2, PLOTTING (VERTICAL AXIS FULL PAGE)      LSWAVE.90
C                     ILR(2,J) = 1, PRINTING                                LSWAVE.91
C                       J = 1, SURFACE ELEVATION                            LSWAVE.92
C                       J = 2, VELOCITY                                     LSWAVE.93
C                       J = 3, OVER-TOPPING RATE                            LSWAVE.94
C       IMJ         = MAP(I-1,J)                                            LSWAVE.95
C       IOB(*)      ROW NO. OF OPEN BOUNDARY SEGMENT NORMAL TO X-AXIS       LSWAVE.96
C       IPJ         = MAP(I+1,J)                                            LSWAVE.97
C       IPLTF       IF .NE. 0, ALL HISTORIES ARE WRITTEN TO TAPE2 FOR       LSWAVE.98
C                     EXTERNAL PLOTTING                                     LSWAVE.99
C       IS          NUMBER OF TIME STEPS BETWEEN STORED HISTORY VALUES      LSWAVE.100
C       ISP         ISP*IS = NUMBER OF TIME STEPS BETWEEN PRINTED           LSWAVE.101
C                     HISTORY VALUES                                        LSWAVE.102
C       IWEIR(*)    ROW NO. OF DAM SEGMENT NORMAL TO X-AXIS                 LSWAVE.103
C       I1(*)       BEGINNING ROW NO. OF COMPUTATIONAL SEGMENT NORMAL       LSWAVE.104
C                     TO X-AXIS                                             LSWAVE.105
C       I1K         = I1(K)                                                 LSWAVE.106
C       I1OB(*)     BEGINNING ROW NO. OF OPEN BOUNDARY SEGMENT NORMAL       LSWAVE.107
C                     TO X-AXIS                                             LSWAVE.108
C       I2WEIR(*)   BEGINNING ROW NO. OF DAM SEGMENT NORMAL TO X-AXIS       LSWAVE.109
C       I2(*)       ENDING ROW NO. OF COMPUTATIONAL SEGMENT NORMAL          LSWAVE.110
C                     TO X-AXIS                                             LSWAVE.111
C       I2K         = I2(K)                                                 LSWAVE.112
C       I2OB(*)     ENDING ROW NO. OF OPEN BOUNDY SEGMENT NORMAL            LSWAVE.113
C                     TO X-AXIS                                             LSWAVE.114
C       I2WEIR(*)   ENDING ROW NO. OF DAM SEGMENT NORMAL TO X-AXIS          LSWAVE.115
```

```
C       I9          DUMMY INTEGER VARIABLE                                  LSWAVE.116
C       J           COLUMN INDEX                                            LSWAVE.117
C       JES(*)      ONE-D INDEX OF POINT FOR WHICH A HISTORY IS STORED      LSWAVE.118
C       JHIST(*)    J-COORDINATE OF POINT FOR WHICH A HISTORY IS STORED     LSWAVE.119
C       JJ1         DUMMY INTEGER VARIABLE                                  LSWAVE.120
C       JJ2         DUMMY INTEGER VARIABLE                                  LSWAVE.121
C       JK(*)       COLUMN NO. OF COMPUTATIONAL SEGMENT NORMAL TO Y-AXIS    LSWAVE.122
C       JOB(*)      COLUMN NO. OF OPEN BOUNDARY SEGMENT NORMAL TO Y-AXIS    LSWAVE.123
C       JWEIR(*)    COLUMN NO. OF DAM SEGMENT NORMAL TO Y-AXIS              LSWAVE.124
C       J1(*)       BEGINNING COLUMN NO. OF COMPUTATIONAL SEGMENT NORMAL    LSWAVE.125
C                     TO Y-AXIS                                             LSWAVE.126
C       J1K         = J1(K)                                                 LSWAVE.127
C       J1OB(*)     BEGINNING COLUMN NO. OF OPEN BOUNDARY SEGMENT NORMAL    LSWAVE.128
C                     TO Y-AXIS                                             LSWAVE.129
C       J1WEIR(*)   BEGINNING COLUMN NO. OF DAM SEGMENT NORMAL TO Y-AXIS    LSWAVE.130
C       J2(*)       ENDING COLUMN NO. OF COMPUTATIONAL SEGMENT NORMAL       LSWAVE.131
C                     TO Y-AXIS                                             LSWAVE.132
C       J2K         = J2(K)                                                 LSWAVE.133
C       J2OB(*)     ENDING COLUMN NO. OF OPEN BOUNDARY SEGMENT NORMAL       LSWAVE.134
C                     TO Y-AXIS                                             LSWAVE.135
C       J2WEIR(*)   ENDING COLUMN NO. OF DAM SEGMENT NORMAL TO Y-AXIS       LSWAVE.136
C       J8          DUMMY INDEX                                             LSWAVE.137
C       J9          DUMMY INDEX                                             LSWAVE.138
C       K           DUMMY COUNTER                                           LSWAVE.139
C       KES         COUNTER FOR HISTORIES                                   LSWAVE.140
C       KNM         = N*M                                                   LSWAVE.141
C       KS          SNAPSHOT COUNTER                                        LSWAVE.142
C       KT          TIME STEP COUNTER                                       LSWAVE.143
C       KVERS       IDENTIFIER FOR CURRENT VERSION OF PROGRAM               LSWAVE.144
C       K9          DUMMY COUNTER                                           LSWAVE.145
C       L           DUMMY COUNTER                                           LSWAVE.146
C       LSKEY       TYPE OF LANDSLIDE MODEL...                              LSWAVE.147
C                     = 1, NO LANDSLIDE                                     LSWAVE.148
C                     = 2, FORCED LANDSLIDE                                 LSWAVE.149
C                     = 3, NATURAL LANDSLIDE                                LSWAVE.150
C                     = 4, BOTTOM MOVEMENT DUE TO FAULT                     LSWAVE.151
C                     = 5, USER ROUTINE                                     LSWAVE.152
C       LSF         IF 0, SNAPSHOTS OF SURFACE ELEVATION NOT DESIRED        LSWAVE.153
```

```
C     LSU        IF 0, SNAPSHOTS OF VELOCITY NOT DESIRED                LSWAVE.154
C     M          TOTAL NUMBER OF GRID POINTS IN X-DIRECTION             LSWAVE.155
C     MAP(*)     MAP TO CONVERT TWO-D INDICES TO ONE-D                  LSWAVE.156
C     MAXCP      MAXIMUM NUMBER OF COMPUTATIONAL POINTS                 LSWAVE.157
C     MAXES      DIMENSION OF ES,US,VS (LIMIT OF NES*MES)               LSWAVE.158
C     MAXES1     DIMENSION OF Q (LIMIT OF NES)                          LSWAVE.159
C     MAXES2     DIMENSION OF JES,IHIST,JHIST (LIMIT OF MES)            LSWAVE.160
C     MB(*)      TWO-DIGIT INTEGER INDICATING TYPE OF BOUNDARY FOR      LSWAVE.161
C                  SEGMENT NORMAL TO X-AXIS ...                         LSWAVE.162
C                    UNIT DIGIT - TYPE AT UPPER BOUNDARY                LSWAVE.163
C                    TENS DIGIT - TYPE AT LOWER BOUNDARY                LSWAVE.164
C                    DIGIT = 0  - OPEN BOUNDARY                         LSWAVE.165
C                    DIGIT = 1  - PARTIALLY REFLECTIVE BOUNDARY         LSWAVE.166
C                    DIGIT = 5  - DAM                                   LSWAVE.167
C     MAXCPM     = MAXCP-1                                              LSWAVE.168
C     MAXCPN     = MAXCPM+N                                             LSWAVE.169
C     MES        NUMBER OF POINTS FOR WHICH HISTORIES ARE STORED        LSWAVE.170
C     MNOW       TOTAL NUMBER OF OPEN BOUNDARY AND DAM SEGMENTS         LSWAVE.171
C     MOB        NUMBER OF OPEN BOUNDARY SEGMENTS NORMAL TO X-AXIS      LSWAVE.172
C     MP         = M  (USED TO PASS DIMENSIONS OF MAP(*) TO SUBPROGRAMS)LSWAVE.173
C     MS         NUMBER OF COMPUTATIONAL SEGMENTS NORMAL TO X-AXIS      LSWAVE.174
C     MWEIR      NUMBER OF DAM SEGMENTS NORMAL TO X-AXIS                LSWAVE.175
C     N          TOTAL NUMBER OF GRID POINTS IN Y-DIRECTION             LSWAVE.176
C     NB(*)      TWO-DIGIT INTEGER INDICATING TYPE OF BOUNDARY FOR      LSWAVE.177
C                  SEGMENT NORMAL TO Y-AXIS ...  (SEE MB(*))            LSWAVE.178
C     NES        NUMBER OF TIME STEPS STORED IN HISTORIES               LSWAVE.179
C     NLP        LINE COUNTER FOR OUTPUT FILE                           LSWAVE.180
C     NMES       = NES*MES                                              LSWAVE.181
C     NOB        NUMBER OF OPEN BOUNDARY SEGMENTS NORMAL TO Y-AXIS      LSWAVE.182
C     NP         = N  (USED TO PASS DIMENSIONS OF MAP(*) TO SUBPROGRAMS)LSWAVE.183
C     NS         NUMBER OF COMPUTATIONAL SEGMENTS NORMAL TO Y-AXIS      LSWAVE.184
C     NUM(*)     INTEGERS 0 THROUGH 9 IN HOLLERITH FORMAT               LSWAVE.185
C     NWEIR      NUMBER OF DAM SEGMENTS NORMAL TO Y-AXIS                LSWAVE.186
C     PI         = 3.141592654                                          LSWAVE.187
C     Q(*)       HISTORY OF DISCHARGE OVER DAM (OVER-TOPPING)           LSWAVE.188
C     QT         TOTAL VOLUME OVER-TOPPING                              LSWAVE.189
C     START      DUMMY LOGICAL VARIABLE                                 LSWAVE.190
C     T          TIME (SECONDS)                                         LSWAVE.191
```

```
C       TMAX        ENDING TIME (TMAX-TZERO = TOTAL SIMULATION TIME)        LSWAVE.192
C       TS(*)       TIMES FOR SNAPSHOTS                                     LSWAVE.193
C       TZERO       STARTING TIME                                           LSWAVE.194
C       T6          DUMMY REAL VARIABLE                                     LSWAVE.195
C       T7          DUMMY REAL VARIABLE                                     LSWAVE.196
C       T9          DUMMY REAL VARIABLE                                     LSWAVE.197
C       U(*)        X-DIRECTION VELOCITY                                    LSWAVE.198
C       UIJ         = U(I,J)                                                LSWAVE.199
C       UM          MEAN VALUE OF U                                         LSWAVE.200
C       UMAX        IF .GT. 0., RANGE USED FOR VELOCITY PLOTS IS            LSWAVE.201
C                     -UMAX TO +UMAX                                        LSWAVE.202
C       UN(*)       U FOR THE NEW TIME STEP                                 LSWAVE.203
C       US(*)       HISTORIES OF VELOCITY (U)                               LSWAVE.204
C       V(*)        Y-DIRECTION VELOCITY                                    LSWAVE.205
C       VIJ         = V(I,J)                                                LSWAVE.206
C       VM          MEAN VALUE OF V                                         LSWAVE.207
C       VN(*)       V FOR THE NEW TIME STEP                                 LSWAVE.208
C       VS(*)       HISTORIES OF VELOCITY (V)                               LSWAVE.209
C       Z0          INITIAL SURFACE ELEVATION                               LSWAVE.210
C                                                                           LSWAVE.211
C                                                                           LSWAVE.212
C ***********************************************************************VARFOR.2
C **  LIST OF INPUT VARIABLES FOR FORCED LANDSLIDE MODEL                 **VARFOR.3
C ***********************************************************************VARFOR.4
C                                                                           VARFOR.5
C       TITLE(*)    SEVENTY CHARACTER SUBTITLE PRINTED WITH SLIDE DATA      VARFOR.6
C       NSL         NUMBER OF SLICES IN LANDSLIDE                           VARFOR.7
C       NXH         NUMBER OF HEIGHTS SPECIFIED FOR EACH SLICE              VARFOR.8
C       NTV         NUMBER OF VELOCITIES SPECIFIED FOR EACH SLICE           VARFOR.9
C       MSUB        NUMBER OF SUBDIVISIONS PER GRID ELEMENT IN THE          VARFOR.10
C                     X-DIRECTION FOR LANDSLIDE CALCULATIONS                VARFOR.11
C       NSUB        NUMBER OF SUBDIVISIONS PER GRID ELEMENT IN THE          VARFOR.12
C                     Y-DIRECTION FOR LANDSLIDE CALCULATIONS                VARFOR.13
C       THETA       FORWARD DIRECTION OF SLIDE (DEGREES COUNTERCLOCKWISE    VARFOR.14
C                     FROM X-AXIS)                                          VARFOR.15
C       WSL         WIDTH OF SLIDE (DISTANCE UNITS)                         VARFOR.16
C       X0          X-COORDINATE (GRID UNITS, NOT DISTANCE) OF RIGHT        VARFOR.17
C                     LEADING CORNER OF SLIDE (-RIGHT- IS DETERMINED AS IF  VARFOR.18
```

```
C                 FACING IN THE DIRECTION OF FORWARD MOTION)          VARFOR.19
C     YO        Y-COORDINATE OF RIGHT LEADING CORNER OF SLIDE         VARFOR.20
C     CDP       PRESSURE DRAG COEFFICIENT (DEFAULTS TO 0.8 IF .LT. 0.) VARFOR.21
C     CDV       VISCOUS DRAG COEFFICIENT (DEFAULTS TO 0.008 IF .LT. 0.)VARFOR.22
C     TSLCNV    FACTOR TO CONVERT TSL(*) TO SECONDS                   VARFOR.23
C     VSLCNV    FACTOR TO CONVERT VSL(*) TO APPROPRIATE UNITS         VARFOR.24
C     XSL(*)    LOCATIONS WHERE SLIDE HEIGHT IS SPECIFIED (DISTANCES  VARFOR.25
C                 FROM LEADING EDGE)                                  VARFOR.26
C     HSL(I,J)  HEIGHT OF SLICE J AT DISTANCE XSL(I) FROM LEADING EDGE VARFOR.27
C     TSL(*)    TIMES AT WHICH SLIDE VELOCTIY IS SPECIFIED            VARFOR.28
C     VSL(I,J)  VELOCITY OF SLICE J AT TIME TSL(I)                    VARFOR.29
C                                                                     VARFOR.30
C                                                                     LSWAVE.214
C ***********************************************************************VARNAT.2
C **  LIST OF INPUT VARIABLES FOR NATURAL LANDSLIDE MODEL            **VARNAT.3
C ***********************************************************************VARNAT.4
C                                                                     VARNAT.5
C     TITLE(*)  SEVENTY CHARACTER SUBTITLE PRINTED WITH SLIDE DATA    VARNAT.6
C     NSL       (UNUSED)                                              VARNAT.7
C     NXH       NUMBER OF HEIGHTS SPECIFIED FOR EACH SLICE            VARNAT.8
C     NTV       (UNUSED)                                              VARNAT.9
C     MSUB      NUMBER OF SUBDIVISIONS PER GRID ELEMENT IN THE        VARNAT.10
C                 X-DIRECTION FOR LANDSLIDE CALCULATIONS              VARNAT.11
C     NSUB      NUMBER OF SUBDIVISIONS PER GRID ELEMENT IN THE        VARNAT.12
C                 Y-DIRECTION FOR LANDSLIDE CALCULATIONS              VARNAT.13
C     THETA     FORWARD DIRECTION OF SLIDE (DEGREES COUNTERCLOCKWISE  VARNAT.14
C                 FROM X-AXIS)                                        VARNAT.15
C     WSL       WIDTH OF SLIDE (DISTANCE UNITS)                       VARNAT.16
C     XO        X-COORDINATE (GRID UNITS, NOT DISTANCE) OF RIGHT      VARNAT.17
C                 LEADING CORNER OF SLIDE (-RIGHT- IS DETERMINED AS IF VARNAT.18
C                 FACING IN THE DIRECTION OF FORWARD MOTION)          VARNAT.19
C     YO        Y-COORDINATE OF RIGHT LEADING CORNER OF SLIDE         VARNAT.20
C     CDP       PRESSURE DRAG COEFFICIENT (DEFAULTS TO 0.8 IF .LT. 0.) VARNAT.21
C     CDV       VISCOUS DRAG COEFFICIENT (DEFAULTS TO 0.008 IF .LT. 0.)VARNAT.22
C     SGSL      SPECIFIC GRAVITY OF SLIDE (DEFAULTS TO 1.7 IF .LT. 0.) VARNAT.23
C     AK        ADDED MASS COEFFICIENT (DEFAULTS TO 0.2 IF .LT. 0.)   VARNAT.24
C     PHI       FRICTION ANGLE (DEGREES) (DEFAULTS TO 10. IF .LT. 0.) VARNAT.25
C     XSL(*)    LOCATIONS WHERE SLIDE HEIGHT IS SPECIFIED (DISTANCES  VARNAT.26
```

```
C                 FROM LEADING EDGE)                                    VARNAT.27
C     HSL(I)     HEIGHT OF SLIDE AT DISTANCE XSL(I) FROM LEADING EDGE   VARNAT.28
C                 (HSL*WSL SHOULD REPRESENT THE CROSS-SECTIONAL AREA)   VARNAT.29
C                                                                       VARNAT.30
C                                                                       LSWAVE.216
C ***********************************************************************VARFLT.2
C **  LIST OF INPUT VARIABLES FOR BOTTOM FAULT MODEL                   **VARFLT.3
C ***********************************************************************VARFLT.4
C                                                                       VARFLT.5
C     TITLE(*)   SEVENTY CHARACTER SUBTITLE PRINTED WITH FAULT DATA     VARFLT.6
C     FI(*)      X-GRID COORDINATES OF TWO POINTS ON FAULT LINE         VARFLT.7
C                  DISPLACED REGION IS TO THE RIGHT OF THE LINE         VARFLT.8
C                  (-RIGHT- IS DETERMINED AS IF STANDING AT THE FIRST   VARFLT.9
C                   POINT, FACING THE SECOND)                           VARFLT.10
C     FJ(*)      Y-GRID COORDINATES FOR THE TWO POINTS DESCRIBING       VARFLT.11
C                  THE FAULT LINE                                       VARFLT.12
C     DFLT       MAGNITUDE OF DISPLACEMENT                              VARFLT.13
C                                                                       LSWAVE.218
C                                                                       LSWAVE.219
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),       //.2
     A      H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                 //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A      HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,ZO,              /DATA/.3
     B        IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTR,              /DATA/.4
     C      I1OB(4),I2OB(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),        /DATA/.5
     D      JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),        /DATA/.6
     E      J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR     /DATA/.7
      COMMON /SEGM/ IK(300),I1(300),I2(300),JK(300),J1(300),J2(300),    /SEGM/.2
     A      MB(300),NB(300)                                             /SEGM/.3
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A      DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,    /SIMU/.3
     B      GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C      KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                 /SIMU/.5
      COMMON /HIST/ ES(2500),US(2500),VS(2500),Q(500),SS(500)           /HIST/.2
      COMMON /DIMS/ MAXES,MAXES1,MAXES2,IDIMD,IDIME,IDIMEH,IDIMOB,      /DIMS/.2
     A      IDIMTS,IDIMWE                                               /DIMS/.3
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,          /DSLD/.2
     A      WSL,DSL,XO,YO,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),            /DSLD/.3
```

```
     B      XSL(101),TSL(101),HSL(101,10),VSL(101,10)                  /DSLD/.4
      COMMON /SLD1/ NSP(10),NKSP,IKSP(500),JKSP(500),XSN(10),VSN(10),   /SLD1/.2
     A      MA,NA,DXA,DYA,DXDXA,DYDXA,DXDYA,DYDYA,AMN,MNA,ITV,LSMAX(10), /SLD1/.3
     B      XA(2500)                                                   /SLD1/.4
      COMMON /SLD2/ XSP(500),LSP(500)                                   /SLD2/.2
      COMMON /SLD3/ HS(1350),HSN(1350),VSA(1350),ICV(1350),ICVN(1350)   /SLD3/.2
      COMMON /SLD4/ NSEG,NSS,KIJ,DS,VTOTAL,HMAXO,XNN(100),HNN(100),     /SLD4/.2
     A      DN(100),VOL(100)                                           /SLD4/.3
      COMMON /SDEF/ KDEF(10),ADEF(10),VDEF(10)                          /SDEF/.2
      DIMENSION MAP(2010)                                              LSWAVE.232
      DATA MAXES,MAXES1,MAXES2,IDIMD,IDIME,IDIMEH,IDIMOB,IDIMTS,IDIMWE  LSWAVE.233
     A    / 2500,   500,    50, 2000, 1350,  1300,     4,    21,     9/ LSWAVE.234
      CALL DATARDR                                                     LSWAVE.235
      CALL DATAPRT                                                     LSWAVE.236
      CALL INITIAL(MAP,M,N)                                            LSWAVE.237
      LSKEYX=LSKEY                                                     LSWAVE.238
      IF (LSKEY .EQ. 2) CALL INITFOR(MAP,M,N)                          LSWAVE.239
      IF (LSKEY .EQ. 3) CALL INITNAT(MAP,M,N)                          LSWAVE.240
      IF (LSKEY .EQ. 4) CALL INITFLT(MAP,M,N)                          LSWAVE.241
      CALL SIMULAT(MAP,M,N,ES,US,VS,Q,NES,MES)                         LSWAVE.242
      LSKEY=LSKEYX                                                     LSWAVE.243
      IF (IENV .NE. 0) CALL SHOTPRT(EH,D,MAXCP,MAP,M,N,T,3,            LSWAVE.244
     A      25HMAXIMUM SURFACE ELEVATION,0)                            LSWAVE.245
      IF ((LSKEY .EQ. 2) .OR. (LSKEY .EQ. 3)) CALL SHOTPRT(D,D,MAXCP,  LSWAVE.246
     A      MAP,M,N,T,2,11HFINAL DEPTH,0)                              LSWAVE.247
      CALL HISTPRT(ES,US,VS,Q,SS,NES,MES)                              LSWAVE.248
C                                                                      LSWAVE.249
C ***** WARNING *****  -PLOTPRT- DESTROYS BLANK COMMON                 LSWAVE.250
      CALL PLOTPRT(ES,NES,MES,EMAX,0,ILR(1,1),2,                       LSWAVE.251
     A      17HSURFACE ELEVATION)                                      LSWAVE.252
      CALL PLOTPRT(US,NES,MES,UMAX,0,ILR(1,2),3,                       LSWAVE.253
     A      24HX-DIRECTION VELOCITY (U))                               LSWAVE.254
      CALL PLOTPRT(VS,NES,MES,UMAX,0,ILR(1,2),3,                       LSWAVE.255
     A      24HY-DIRECTION VELOCITY (V))                               LSWAVE.256
      CALL PLOTPRT(Q,NES,1,0.,1,ILR(1,3),2,                            LSWAVE.257
     A      17HOVER-TOPPING RATE)                                      LSWAVE.258
      QT=QT*DT                                                         LSWAVE.259
      PRINT 101,QT                                                     LSWAVE.260
```

```
  101 FORMAT (1H1,27HTOTAL VOLUME OVER-TOPPING =,1PE15.8)                LSWAVE.261
C                                                                        LSWAVE.262
C     STORE HYDROGRAPH (OVER THE DAM) INTO TAPE1, TO BE USED FOR THE     LSWAVE.263
C     DOWNSTREAM FLOOD ROUTING MODEL                                     LSWAVE.264
      WRITE (1) TZERO,DT*IS,NES,(Q(I),I=1,NES)                           LSWAVE.265
C                                                                        LSWAVE.266
      IF (IPLTF .EQ. 0) GOTO 10                                          LSWAVE.267
      IF (MES .EQ. 0) GOTO 10                                            LSWAVE.268
C                                                                        LSWAVE.269
C     STORE DATA TO TAPE2 FOR EXTERNAL PLOTTING                          LSWAVE.270
      WRITE (2) TZERO,DT*IS,NES,MES,(IHIST(I),JHIST(I),I=1,MES)          LSWAVE.271
      NMES=NES*MES                                                       LSWAVE.272
      WRITE (2) (ES(I),I=1,NMES)                                         LSWAVE.273
      WRITE (2) (US(I),I=1,NMES)                                         LSWAVE.274
      WRITE (2) (VS(I),I=1,NMES)                                         LSWAVE.275
      WRITE (2) (Q(I),I=1,NES)                                           LSWAVE.276
   10 CONTINUE                                                           LSWAVE.277
      STOP                                                               LSWAVE.278
      END                                                                LSWAVE.279
```

```
      SUBROUTINE DATARDR                                                DATARDR.2
C                                                                       DATARDR.3
C     READ INPUT DATA FOR THE HYDRODYNAMIC SUBMODEL                     DATARDR.4
C                                                                       DATARDR.5
C     ALL INPUT DATA (EXCEPT CMANN) SHOULD USE ONE UNIT SYSTEM          DATARDR.6
C       CONSISTENTLY (FPS, CGS, OR MKS)                                 DATARDR.7
C     CMANN (MANNING N) IS ALWAYS GIVEN IN FPS UNITS                    DATARDR.8
C                                                                       DATARDR.9
C     EXTERNALS REFERENCED                                              DATARDR.10
C          PRT4                                                         DATARDR.11
C          ERR                                                          DATARDR.12
C     EXTERNALS CONTAINING REFERENCES                                   DATARDR.13
C          LSWAVE  (MAIN PROGRAM)                                       DATARDR.14
C                                                                       DATARDR.15
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),       //.2
     A       H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A       HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,             /DATA/.3
     B         IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,             /DATA/.4
     C       I1OB(4),I2OB(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),       /DATA/.5
     D       JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),       /DATA/.6
     E       J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR    /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A       DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,   /SIMU/.3
     B       GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50), /SIMU/.4
     C       KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                /SIMU/.5
      COMMON /DIMS/ MAXES,MAXES1,MAXES2,IDIMD,IDIME,IDIMEH,IDIMOB,      /DIMS/.2
     A       IDIMTS,IDIMWE                                              /DIMS/.3
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,          /DSLD/.2
     A       WSL,DSL,XO,YO,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),           /DSLD/.3
     B       XSL(101),TSL(101),HSL(101,10),VSL(101,10)                  /DSLD/.4
      DATA KVERS/4HA.3 /                                                DATARDR.21
C                                                                       DATARDR.22
      CALL PRT4(KVERS)                                                  DATARDR.23
      READ 101,M,N,MOB,NOB,MWEIR,NWEIR,IC,ICC                           DATARDR.24
      READ 102,TZERO,TMAX,DT,DX,DY,DCONV,Z0                             DATARDR.25
      READ 102,CMANN,CED,CD,G                                           DATARDR.26
      READ 101,MES,IS,ISP,ILR,KS,LSE,LSU,IENV,IPLTF                     DATARDR.27
```

```
      READ 102,EMAX,UMAX                                                DATARDR.28
      DO 10 I=1,M                                                       DATARDR.29
   10 READ 103,(D((I-1)*N+J),J=1,N)                                     DATARDR.30
      IF (MOB .GT. 0) READ 104,(I0B(K),J10B(K),J20B(K),K=1,MOB)         DATARDR.31
      IF (NOB .GT. 0) READ 104,(J0B(K),I10B(K),I20B(K),K=1,NOB)         DATARDR.32
      IF (MWEIR .GT. 0) READ 105,(IWEIR(K),J1WEIR(K),J2WEIR(K),         DATARDR.33
     A      HWEIRI(K),K=1,MWEIR)                                        DATARDR.34
      IF (NWEIR .GT. 0) READ 105,(JWEIR(K),I1WEIR(K),I2WEIR(K),         DATARDR.35
     A      HWEIRJ(K),K=1,NWEIR)                                        DATARDR.36
      IF (MES .GT. 0) READ 101,(IHIST(K),JHIST(K),K=1,MES)              DATARDR.37
      IF (KS .GT. 0) READ 102,(TS(K),K=1,KS)                            DATARDR.38
      READ 106,LSKEY,TITLE                                              DATARDR.39
      KNM=4*N                                                           DATARDR.40
      IF (KNM.GT.IDIMD) CALL ERR( 1,1,5,41HREQUESTED GRID SIZE (M*N) EXCDATARDR.41
     ,EEDS MAXIMUM,4,2,1HM,M,2,1HN,N,2,3HM*N,KNM,2,5HIDIMD,IDIMD)       DATARDR.42
      IF ((LSKEY.LE.0) .OR. (LSKEY.GT.5)) CALL ERR( 2,1,3,24HLSKEY HAS IDATARDR.43
     ,MPROPER VALUE,1,2,5HLSKEY,LSKEY)                                  DATARDR.44
      NES=INT((TMAX-TZERO)/DT+1.)/IS+2                                  DATARDR.45
      IF (MES*NES .GT. MAXES) CALL ERR( 3,1,7, 67H ARRAYS ES, US, VS TODDATARDR.46
     , SMALL FOR REQUESTED NUMBER OF HISTORY POINTS,                    DATARDR.47
     ,      4,2,5HMAXES,MAXES,2,3HMES,MES,2,3HNES,NES,2,2HIS,IS)        DATARDR.48
      IF (NES .GT. MAXES1) CALL ERR( 4,1,2,17HARRAY Q TOO SMALL,        DATARDR.49
     ,  2,2,6HMAXES1,MAXES1,2,3HNES,NES)                                DATARDR.50
      IF (MES .GT. MAXES2) CALL ERR( 5,1,7,69HARRAYS JES, IHIST, JHIST TDATARDR.51
     ,OO SMALL (ALSO CHECK IBL, IPT IN -HISTPRT-),2,2,6HMAXES2,MAXES2,  DATARDR.52
     ,  2,3HMES,MES)                                                    DATARDR.53
      IF (KS .GT. IDIMTS) CALL ERR( 6,1,5,45HREQUESTED NUMBER OF SNAPSHODATARDR.54
     ,TS EXCEEDS MAXIMUM,2,2,2HKS,KS,2,6HIDIMTS,IDIMTS)                 DATARDR.55
      RETURN                                                            DATARDR.56
C                                                                       DATARDR.57
  101 FORMAT (16I5)                                                     DATARDR.58
  102 FORMAT (8F10.5)                                                   DATARDR.59
  103 FORMAT (16F5.0)                                                   DATARDR.60
  104 FORMAT (8(3I3,1X))                                                DATARDR.61
  105 FORMAT (4(3I3,1XF10.5))                                           DATARDR.62
  106 FORMAT (I1,9X7A10)                                                DATARDR.63
      END                                                               DATARDR.64
```

```
      SUBROUTINE DATAPRT                                                DATAPRT.2
C                                                                       DATAPRT.3
C     PRINT INPUT DATA FOR THE HYDRODYNAMIC SUBMODEL                    DATAPRT.4
C                                                                       DATAPRT.5
C     EXTERNALS REFERENCED                                              DATAPRT.6
C          PRT1                                                         DATAPRT.7
C          PRT2                                                         DATAPRT.8
C          PRT3                                                         DATAPRT.9
C          ERR                                                          DATAPRT.10
C     EXTERNALS CONTAINING REFERENCES                                   DATAPRT.11
C          LSWAVE  (MAIN PROGRAM)                                       DATAPRT.12
C                                                                       DATAPRT.13
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),       //.2
     A     H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                  //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A     HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,               /DATA/.3
     B       IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,               /DATA/.4
     C     I1OB(4),I2OB(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),         /DATA/.5
     D     JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),         /DATA/.6
     E     J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR      /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A     DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,     /SIMU/.3
     B     GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C     KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                  /SIMU/.5
      REAL NUM(10)                                                      DATAPRT.17
      DIMENSION D0(2000),D1(2000)                                       DATAPRT.18
      EQUIVALENCE (D0,C),(D1,EH)                                        DATAPRT.19
      DATA NUM/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/                 DATAPRT.20
C                                                                       DATAPRT.21
      PRINT 902                                                         DATAPRT.22
      CALL PRT1(2,1HM,M,5,42HTOTAL NUMBER OF GRID POINTS IN X-DIRECTION)DATAPRT.23
      CALL PRT1(2,1HN,N,5,42HTOTAL NUMBER OF GRID POINTS IN Y-DIRECTION)DATAPRT.24
      CALL PRT1(2,3HMOB,MOB,5,49HNUMBER OF OPEN BOUNDARY SEGMENTS NORMALDATAPRT.25
     , TO X-AXIS)                                                       DATAPRT.26
      CALL PRT1(2,3HNOB,NOB,5,49HNUMBER OF OPEN BOUNDARY SEGMENTS NORMALDATAPRT.27
     , TO Y-AXIS)                                                       DATAPRT.28
      CALL PRT1(2,5HNWEIR,MWEIR,4,39HNUMBER OF DAM SEGMENTS NORMAL TO X-DATAPRT.29
     ,AXIS)                                                             DATAPRT.30
```

```
 CALL PRT1(2,5HNWEIR,NWEIR,4,39HNUMBER OF DAM SEGMENTS NORMAL TO Y-DATAPRT.31
,AXIS)                                                              DATAPRT.32
 IF (IC.EQ.0) CALL PRT1(2,2HIC,IC,3,27H=> CONVECTIVE TERM EXCLUDED)DATAPRT.33
 IF (IC.NE.0) CALL PRT1(2,2HIC,IC,3,27H=> CONVECTIVE TERM INCLUDED)DATAPRT.34
 IF (ICC.EQ.0) CALL PRT1(2,3HICC,ICC,3,29H=> CONTINUITY EQUATION LIDATAPRT.35
,NEAR)                                                              DATAPRT.36
 IF (ICC.NE.0) CALL PRT1(2,3HICC,ICC,4,33H=> CONTINUITY EQUATION NODATAPRT.37
,N-LINEAR)                                                          DATAPRT.38
 CALL PRT1(1,5HTZERO,TZERO,3,24HBEGINNING TIME (SECONDS))           DATAPRT.39
 CALL PRT1(1,4HTMAX,TMAX,3,21HENDING TIME (SECONDS))                DATAPRT.40
 CALL PRT1(1,2HDT,DT,2,19HTIME STEP (SECONDS))                      DATAPRT.41
 CALL PRT1(1,2HDX,DX,3,27HGRID SPACING IN X-DIRECTION)              DATAPRT.42
 CALL PRT1(1,2HDY,DY,3,27HGRID SPACING IN Y-DIRECTION)              DATAPRT.43
 CALL PRT1(1,5HDCONV,DCONV,3,28HCONVERSION FACTOR FOR DEPTHS)       DATAPRT.44
 IF (ZG.EQ.0.) CALL PRT1(1,2HZG,ZG,4,39H=> INPUT GRID IN DEPTHS, NODATAPRT.45
,T ELEVATIONS)                                                      DATAPRT.46
 IF (ZO.NE.0.) CALL PRT1(1,2HZO,ZO,3,25HINITIAL SURFACE ELEVATION) DATAPRT.47
 CALL PRT1(1,5HCMANN,CMANN,3,29HMANNING ROUGHNESS COEFFICIENT)      DATAPRT.48
 CALL PRT1(1,3HCED,CED,7,66HCOEFFICIENT OF ENERGY DISSIPATION AT PADATAPRT.49
,RTIALLY REFLECTIVE BOUNDARY)                                       DATAPRT.50
 CALL PRT1(1,2HCD,CD,4,38HDISCHARGE COEFFICIENT FOR WEIR FORMULA)  DATAPRT.51
 CALL PRT1(1,1HG,G,3,26HGRAVITATIONAL ACCELERATION)                 DATAPRT.52
 PRINT 913                                                          DATAPRT.53
 IF (MES.EQ.0) CALL PRT1(2,3HMES,MES,4,35H=> ONLY OVER-TOPPING HISTDATAPRT.54
,ORY STORED)                                                        DATAPRT.55
 IF (MES.NE.0) CALL PRT1(2,3HMES,MES,5,47HNUMBER OF POINTS FOR WHICDATAPRT.56
,H HISTORIES ARE STORED)                                            DATAPRT.57
 CALL PRT1(2,2HIS,IS,5,50HNUMBER OF TIME STEPS BETWEEN STORED HISTODATAPRT.58
,RY VALUES)                                                         DATAPRT.59
 IL=1                                                               DATAPRT.60
 IF ((ILR(2,1).EQ.0).AND.(ILR(2,2).EQ.0).AND.(ILR(2,3).EQ.0)) IL=0 DATAPRT.61
 IF (IL.EQ.0) CALL PRT1(2,3HISP,ISP,4,31H(UNUSED - NO HISTORIES PRIDATAPRT.62
,NTED))                                                             DATAPRT.63
 IF (IL.NE.0) CALL PRT1(2,3HISP,ISP,6,60HISP*IS = NUMBER OF TIME STDATAPRT.64
,EPS BETWEEN PRINTED HISTORY VALUES)                                DATAPRT.65
 IF (ILR(1,1).EQ.0) CALL PRT1(2,8HILR(1,1),ILR(1,1),5,42H=> NO PLOTDATAPRT.66
,S OF SURFACE ELEVATION HISTORIES)                                  DATAPRT.67
 IF ((ILR(1,1).NE.0).AND.(ILR(1,1).NE.2)) CALL PRT1(2,8HILR(1,1),  DATAPRT.68
```

```
, ILR(1,1),5,49H=> HALF PAGE PLOTS OF SURFACE ELEVATION HISTORIES) DATAPRT.69
 IF (ILR(1,1).EQ.2) CALL PRT1(2,8HILR(1,1),ILR(1,1),5,49H=> FULL PADATAPRT.70
,GE PLOTS OF SURFACE ELEVATION HISTORIES)                          DATAPRT.71
 IF (ILR(2,1).EQ.0) CALL PRT1(2,8HILR(2,1),ILR(2,1),5,42H=> SURFACEDATAPRT.72
, ELEVATION HISTORIES NOT PRINTED)                                 DATAPRT.73
 IF (ILR(2,1).NE.0) CALL PRT1(2,8HILR(2,1),ILR(2,1),4,38H=> SURFACEDATAPRT.74
, ELEVATION HISTORIES PRINTED)                                     DATAPRT.75
 IF (ILR(1,2).EQ.0) CALL PRT1(2,8HILR(1,2),ILR(1,2),4,33H=> NO PLOTDATAPRT.76
,S OF VELOCITY HISTORIES)                                          DATAPRT.77
 IF ((ILR(1,2).NE.0).AND.(ILR(1,2).NE.2)) CALL PRT1(2,8HILR(1,2),  DATAPRT.78
, ILR(1,2),4,40H=> HALF PAGE PLOTS OF VELOCITY HISTORIES)          DATAPRT.79
 IF (ILR(1,2).EQ.2) CALL PRT1(2,8HILR(1,2),ILR(1,2),4,40H=> FULL PADATAPRT.80
,GE PLOTS OF VELOCITY HISTORIES)                                   DATAPRT.81
 IF (ILR(2,2).EQ.0) CALL PRT1(2,8HILR(2,2),ILR(2,2),4,33H=> VELOCITDATAPRT.82
,Y HISTORIES NOT PRINTED)                                          DATAPRT.83
 IF (ILR(2,2).NE.0) CALL PRT1(2,8HILR(2,2),ILR(2,2),3,29H=> VELOCITDATAPRT.84
,Y HISTORIES PRINTED)                                              DATAPRT.85
 IF (ILR(1,3).EQ.0) CALL PRT1(2,8HILR(1,3),ILR(1,3),4,39H=> NO PLOTDATAPRT.86
, OF OVER-TOPPING RATE HISTORY)                                    DATAPRT.87
 IF ((ILR(1,3).NE.0).AND.(ILR(1,3).NE.2)) CALL PRT1(2,8HILR(1,3),  DATAPRT.88
, ILR(1,3),5,46H=> HALF PAGE PLOT OF OVER-TOPPING RATE HISTORY)    DATAPRT.89
 IF (ILR(1,3).EQ.2) CALL PRT1(2,8HILR(1,3),ILR(1,3),5,46H=> FULL PADATAPRT.90
,GE PLOT OF OVER-TOPPING RATE HISTORY)                             DATAPRT.91
 IF (ILR(2,3).EQ.0) CALL PRT1(2,8HILR(2,3),ILR(2,3),4,35H=> OVER-TODATAPRT.92
,PPING HISTORY NOT PRINTED)                                        DATAPRT.93
 IF (ILR(2,3).NE.0) CALL PRT1(2,8HILR(2,3),ILR(2,3),4,31H=> OVER-TODATAPRT.94
,PPING HISTORY PRINTED)                                            DATAPRT.95
 IL=1                                                              DATAPRT.96
 IF ((LSE.EQ.0).AND.(LSU.EQ.0)) IL=0                               DATAPRT.97
 IF (IL.EQ.0) CALL PRT1(2,2HKS,KS,3,23H(UNUSED - NO SNAPSHOTS))    DATAPRT.98
 IF (IL.NE.0) CALL PRT1(2,2HKS,KS,4,38HNUMBER OF SNAPSHOTS SPECIFIEDATAPRT.99
,D IN TS(*))                                                       DATAPRT.100
 IF (LSE.EQ.0) CALL PRT1(2,3HLSE,LSE,4,36H=> NO SNAPSHOTS OF SURFACDATAPRT.101
,E ELEVATION)                                                      DATAPRT.102
 IF (LSE.EQ.1) CALL PRT1(2,3HLSE,LSE,7,69H=> SNAPSHOTS OF SURFACE EDATAPRT.103
,LEVATION PRINTED FOR SPECIFIED TIMES (TS(*)))                     DATAPRT.104
 IF (LSU.EQ.0) CALL PRT1(2,3HLSU,LSU,3,27H=> NO SNAPSHOTS OF VELOCIDATAPRT.105
,TY)                                                               DATAPRT.106
```

```
      IF (LSU.EQ.1) CALL PRT1(2,3HLSU,LSU,6,60H=> SNAPSHOTS OF VELOCITY DATAPRT.107
     ,PRINTED FOR SPECIFIED TIMES (TS(*)))                              DATAPRT.108
      IF (IENV.EQ.0) CALL PRT1(2,4HIENV,IENV,4,40H=> MAXIMUM SURFACE ELEDATAPRT.109
     ,VATION NOT PRINTED)                                               DATAPRT.110
      IF (IENV.NE.0) CALL PRT1(2,4HIENV,IENV,4,36H=> MAXIMUM SURFACE ELEDATAPRT.111
     ,VATION PRINTED)                                                   DATAPRT.112
      IF (IPLTF.EQ.0) CALL PRT1(2,5HIPLTF,IPLTF,3,23H=> NO PLOT FILE WRIDATAPRT.113
     ,TTEN)                                                             DATAPRT.114
      IF (IPLTF.NE.0) CALL PRT1(2,5HIPLTF,IPLTF,4,34H=> FILE WRITTEN FORDATAPRT.115
     , EXTERNAL PLOTS)                                                  DATAPRT.116
      IF ((ILR(1,1).NE.0).AND.(EMAX.EQ.0.)) CALL PRT1(1,4HEMAX,EMAX,5,  DATAPRT.117
     , 50H=> RANGE FOR ELEVATION PLOTS DETERMINED INTERNALLY)           DATAPRT.118
      IF ((ILR(1,1).NE.0).AND.(EMAX.NE.0.)) CALL PRT1(1,4HEMAX,EMAX,4,  DATAPRT.119
     , 33HMAXIMUM RANGE FOR ELEVATION PLOTS)                            DATAPRT.120
      IF (ILR(1,1).EQ.0) CALL PRT1(1,4HEMAX,EMAX,3,29H(UNUSED - NO ELEVADATAPRT.121
     ,TION PLOTS))                                                      DATAPRT.122
      IF ((ILR(1,2).NE.0).AND.(UMAX.EQ.0.)) CALL PRT1(1,4HUMAX,UMAX,5,  DATAPRT.123
     , 49H=> RANGE FOR VELOCITY PLOTS DETERMINED INTERNALLY)            DATAPRT.124
      IF ((ILR(1,2).NE.0).AND.(UMAX.NE.0.)) CALL PRT1(1,4HUMAX,UMAX,4,  DATAPRT.125
     , 32HMAXIMUM RANGE FOR VELOCITY PLOTS)                             DATAPRT.126
      IF (ILR(1,2).EQ.0) CALL PRT1(1,4HUMAX,UMAX,3,28H(UNUSED - NO VELOCDATAPRT.127
     ,ITY PLOTS))                                                       DATAPRT.128
C                                                                       DATAPRT.129
      KNM=M*N                                                           DATAPRT.130
      IF (Z0.NE.0.) GOTO 10                                             DATAPRT.131
      IF (DCONV.NE.1.) GOTO 5                                           DATAPRT.132
      CALL PRT2(D,N,M,5HDEPTH)                                          DATAPRT.133
      DO 3 K=1,KNM                                                      DATAPRT.134
    3 D0(K)=D(K)                                                        DATAPRT.135
      GOTO 20                                                           DATAPRT.136
    5 CALL PRT2(D,N,M,5HINPUT)                                          DATAPRT.137
      DO 8 K=1,KNM                                                      DATAPRT.138
    8 D0(K)=D(K)*DCONV                                                  DATAPRT.139
      CALL PRT2(D0,N,M,5HDEPTH)                                         DATAPRT.140
      GOTO 20                                                           DATAPRT.141
   10 CALL PRT2(D,N,M,5HINPUT)                                          DATAPRT.142
      DO 13 K=1,KNM                                                     DATAPRT.143
      D0(K)=0.                                                          DATAPRT.144
```

```
      IF (D(K) .NE. 0.) D0(K)=ZG-D(K)                                   DATAPRT.145
   13 D0(K)=D0(K)*DCONV                                                 DATAPRT.146
      CALL PRT2(D0,N,N,5HDEPTH)                                         DATAPRT.147
   20 CONTINUE                                                          DATAPRT.148
      MNOW=MOB+NOB+MWEIR+NWEIR                                          DATAPRT.149
      PRINT 901                                                         DATAPRT.150
      NLP=0                                                             DATAPRT.151
      IF (MNOW .EQ. 0) GOTO 60                                          DATAPRT.152
      IF (MOB.EQ.0) GOTO 30                                             DATAPRT.153
      NLP=NLP+8+MOB                                                     DATAPRT.154
      PRINT 904                                                         DATAPRT.155
      PRINT 906                                                         DATAPRT.156
      DO 25 K=1,MOB                                                     DATAPRT.157
      PRINT 907,K,IOB(K),J1OB(K),J2OB(K)                                DATAPRT.158
   25 IF (J1OB(K).GT.J2OB(K)) CALL ERR( 7,1,4,40HJ1OB(K) MUST NOT BE GREDATAPRT.159
     ,ATER THAN J2OB(K),3,2,1HK,K,2,7HJ1OB(K),J1OB(K),2,7HJ2OB(K),      DATAPRT.160
     , J2OB(K))                                                         DATAPRT.161
   30 IF (NOB.EQ.0) GOTO 40                                             DATAPRT.162
      NLP=NLP+8+NOB                                                     DATAPRT.163
      PRINT 905                                                         DATAPRT.164
      PRINT 908                                                         DATAPRT.165
      DO 35 K=1,NOB                                                     DATAPRT.166
      PRINT 907,K,JOB(K),I1OB(K),I2OB(K)                                DATAPRT.167
   35 IF (I1OB(K).GT.I2OB(K)) CALL ERR( 8,1,4,40HI1OB(K) MUST NOT BE GREDATAPRT.168
     ,ATER THAN J2OB(K),3,2,1HK,K,2,7HI1OB(K),I1OB(K),2,7HI2OB(K),      DATAPRT.169
     , I2OB(K))                                                         DATAPRT.170
   40 IF (MWEIR.EQ.0) GOTO 50                                           DATAPRT.171
      NLP=NLP+8+MWEIR                                                   DATAPRT.172
      PRINT 909                                                         DATAPRT.173
      PRINT 911                                                         DATAPRT.174
      DO 45 K=1,MWEIR                                                   DATAPRT.175
      PRINT 907,K,IWEIR(K),J1WEIR(K),J2WEIR(K),HWEIRI(K)                DATAPRT.176
   45 IF (J1WEIR(K).GT.J2WEIR(K)) CALL ERR( 9,1,5,44HJ1WEIR(K) MUST NOT DATAPRT.177
     ,BE GREATER THAN J2WEIR(K),3,2,1HK,K,2,9HJ1WEIR(K),J1WEIR(K),      DATAPRT.178
     , 2,9HJ2WEIR(K),J2WEIR(K))                                         DATAPRT.179
   50 IF (NWEIR.EQ.0) GOTO 60                                           DATAPRT.180
      NLP=NLP+8+NWEIR                                                   DATAPRT.181
      PRINT 910                                                         DATAPRT.182
```

```
      PRINT 912                                                         DATAPRT.183
      DO 55 K=1,NWEIR                                                   DATAPRT.184
      PRINT 907,K,JWEIR(K),I1WEIR(K),I2WEIR(K),HWEIRJ(K)                DATAPRT.185
   55 IF (I1WEIR(K).GT.I2WEIR(K)) CALL ERR(10,1,5,44HI1WEIR(K) MUST NOT DATAPRT.186
     ,BE GREATER THAN I2WEIR(K),3,2,1HK,K,2,9HI1WEIR(K),I1WEIR(K),      DATAPRT.187
     , 2,9HI2WEIR(K),I2WEIR(K))                                         DATAPRT.188
   60 CONTINUE                                                          DATAPRT.189
      DO 70 K=1,KNM                                                     DATAPRT.190
      D1(K)=1H+                                                         DATAPRT.191
      IF (DQ(K) .EQ. 0.) D1(K)=1HD                                      DATAPRT.192
      IF (DQ(K) .LT. 0.) D1(K)=1H-                                      DATAPRT.193
   70 CONTINUE                                                          DATAPRT.194
      IF (MNOW .EQ. 0) GOTO 112                                         DATAPRT.195
      IF (MOB .EQ. 0) GOTO 80                                           DATAPRT.196
      DO 75 K=1,MOB                                                     DATAPRT.197
      I=IOB(K)                                                          DATAPRT.198
      JJ1=J1OB(K)                                                       DATAPRT.199
      JJ2=J2OB(K)                                                       DATAPRT.200
      DO 75 J=JJ1,JJ2                                                   DATAPRT.201
      IJ=(I-1)*N+J                                                      DATAPRT.202
   75 D1(IJ)=1H=                                                        DATAPRT.203
   80 IF (NOB .EQ. 0) GOTO 90                                           DATAPRT.204
      DO 85 K=1,NOB                                                     DATAPRT.205
      J=JOB(K)                                                          DATAPRT.206
      II1=I1OB(K)                                                       DATAPRT.207
      II2=I2OB(K)                                                       DATAPRT.208
      DO 85 I=II1,II2                                                   DATAPRT.209
      IJ=(I-1)*N+J                                                      DATAPRT.210
   85 D1(IJ)=1H=                                                        DATAPRT.211
   90 IF (MWEIR .EQ. 0) GOTO 100                                        DATAPRT.212
      DO 95 K=1,MWEIR                                                   DATAPRT.213
      I=IWEIR(K)                                                        DATAPRT.214
      JJ1=J1WEIR(K)                                                     DATAPRT.215
      JJ2=J2WEIR(K)                                                     DATAPRT.216
      DO 95 J=JJ1,JJ2                                                   DATAPRT.217
      IJ=(I-1)*N+J                                                      DATAPRT.218
   95 D1(IJ)=1H/                                                        DATAPRT.219
  100 IF (NWEIR .EQ. 0) GOTO 110                                        DATAPRT.220
```

```
      DO 105 K=1,NWEIR                                                  DATAPRT.221
      J=JWEIR(K)                                                        DATAPRT.222
      II1=I1WEIR(K)                                                     DATAPRT.223
      II2=I2WEIR(K)                                                     DATAPRT.224
      DO 105 I=II1,II2                                                  DATAPRT.225
      IJ=(I-1)*N+J                                                      DATAPRT.226
  105 D1(IJ)=1H/                                                        DATAPRT.227
  110 CONTINUE                                                          DATAPRT.228
      NLP=NLP+13+M                                                      DATAPRT.229
      IF (NLP .LE. 63) GOTO 111                                         DATAPRT.230
      NLP=13+M                                                          DATAPRT.231
      PRINT 901                                                         DATAPRT.232
  111 PRINT 915                                                         DATAPRT.233
      CALL PRT3(D1,N,M)                                                 DATAPRT.234
      PRINT 903                                                         DATAPRT.235
  112 CONTINUE                                                          DATAPRT.236
      IF (MES .EQ. 0) GOTO 130                                          DATAPRT.237
      DO 125 K=1,MES                                                    DATAPRT.238
      IJ=(IHIST(K)-1)*N+JHIST(K)                                        DATAPRT.239
  125 D1(IJ)=NUM(MOD(K,10)+1)                                           DATAPRT.240
      IF (NLP .NE. 0) PRINT 901                                         DATAPRT.241
      NLP=5+((MES-1)/8+1)+14+M+3                                        DATAPRT.242
      PRINT 917                                                         DATAPRT.243
      PRINT 918,(1H(,IHIST(K),JHIST(K),1H),K=1,MES)                     DATAPRT.244
      PRINT 920                                                         DATAPRT.245
      CALL PRT3(D1,N,M)                                                 DATAPRT.246
      PRINT 903                                                         DATAPRT.247
  130 CONTINUE                                                          DATAPRT.248
      IF (KS .EQ. 0) GOTO 140                                           DATAPRT.249
      NLP=NLP+11+3*(KS/8)                                               DATAPRT.250
      IF (NLP .LE. 63) GOTO 131                                         DATAPRT.251
      NLP=11+3*(KS/8)                                                   DATAPRT.252
      PRINT 901                                                         DATAPRT.253
  131 PRINT 924                                                         DATAPRT.254
      PRINT 925,(TS(K),K=1,KS)                                          DATAPRT.255
      PRINT 926                                                         DATAPRT.256
      DO 135 K=1,KS                                                     DATAPRT.257
  135 ITS(K)=INT((TS(K)-TZERO)/DT+0.5)                                  DATAPRT.258
```

```
      PRINT 927,(ITS(K),K=1,KS)                                         DATAPRT.259
      PRINT 928                                                         DATAPRT.260
      PRINT 925,(ITS(K)*DT+TZERO,K=1,KS)                                DATAPRT.261
  140 CONTINUE                                                          DATAPRT.262
      RETURN                                                            DATAPRT.263
C                                                                       DATAPRT.264
  901 FORMAT (1H1)                                                      DATAPRT.265
  902 FORMAT (11X8HVARIABLE,6X5HVALUE,11X15HDESCRIPTION  OR,            DATAPRT.266
     A      13H  => RESULT  /11X8(1H-),6X8(1H-),8X8D(1H-)/)             DATAPRT.267
  903 FORMAT (3(/))                                                     DATAPRT.268
  904 FORMAT (3(/),33H OPEN BOUNDARIES NORMAL TO X-AXIS/)               DATAPRT.269
  905 FORMAT (3(/),33H OPEN BOUNDARIES NORMAL TO Y-AXIS/)               DATAPRT.270
  906 FORMAT (10X1HK,7X6HI0B(K),5X7HJ10B(K),4X7HJ20B(K)                 DATAPRT.271
     A      /6X9H(K=1,NOB),4X5H(ROW),4X9H(BEG COL),2X9H(END COL)/)      DATAPRT.272
  907 FORMAT (4(9XI2),4XF9.2)                                           DATAPRT.273
  908 FORMAT (10X1HK,7X6HJ0B(K),5X7HI10B(K),4X7HI20B(K)                 DATAPRT.274
     A      /6X9H(K=1,NOB),4X5H(COL),4X9H(BEG ROW),2X9H(END ROW)/)      DATAPRT.275
  909 FORMAT (3(/),30H DAM SEGMENTS NORMAL TO X-AXIS/)                  DATAPRT.276
  910 FORMAT (3(/),30H DAM SEGMENTS NORMAL TO Y-AXIS/)                  DATAPRT.277
  911 FORMAT (10X1HK,6X6HIWEIR(K),3X9HJ1WEIR(K),2X9HJ2WEIR(K),          DATAPRT.278
     A      2X9HHWEIRI(K)/5X11H(K=1,NWEIR),3X5H(ROW),4X9H(BEG COL),     DATAPRT.279
     B      2X9H(END COL),3X8H(HEIGHT)/)                                DATAPRT.280
  912 FORMAT (10X1HK,6X6HJWEIR(K),3X9HI1WEIR(K),2X9HI2WEIR(K),          DATAPRT.281
     A      2X9HHWEIRJ(K)/5X11H(K=1,NWEIR),3X5H(COL),4X9H(BEG ROW),     DATAPRT.282
     B      2X9H(END ROW),3X8H(HEIGHT)/)                                DATAPRT.283
  913 FORMAT (1H )                                                      DATAPRT.284
  915 FORMAT (3(/),44H BOUNDARIES PLOTTED ON COMPUTATIONAL GRID --,     DATAPRT.285
     A      5X9H/  -  DAM                                               DATAPRT.286
     B      /49X19H=  -  OPEN BOUNDARY/49X20H+  -  D(I,J) .GT. 0.       DATAPRT.287
     C      /49X20HD  -  D(I,J) .EQ. 0./49X20H-  -  D(I,J) .LT. 0.//)   DATAPRT.288
  917 FORMAT (3(/),1X17HHISTORY LOCATIONS/)                             DATAPRT.289
  918 FORMAT (8(5XA1,I2,1H,,I2,A1))                                     DATAPRT.290
  920 FORMAT (3(/),48H HISTORY LOCATIONS PLOTTED ON COMPUTATIONAL GRID  DATAPRT.291
     A      //5X50H1, 2, ...  -  HISTORY LOCATIONS (ORDERED AS ABOVE,,  DATAPRT.292
     B      8H MOD 10)/13X20H+  -  D(I,J) .GT. 0.                      DATAPRT.293
     C      /13X20HD  -  D(I,J) .EQ. 0./13X20H-  -  D(I,J) .LT. 0.//)   DATAPRT.294
  924 FORMAT (3(/),44H DESIRED TIMES FOR SNAPSHOTS  (TS(K),K=1,KS)/)    DATAPRT.295
  925 FORMAT ((4X8(2XF10.3)))                                           DATAPRT.296
```

```
926 FORMAT (/24H    -- NEAREST TIME STEPS/)                          DATAPRT.297
927 FORMAT ((8(8XI4)))                                               DATAPRT.298
928 FORMAT (/32H    -- ACTUAL TIMES FOR SNAPSHOTS/)                  DATAPRT.299
    END                                                              DATAPRT.300
```

```
      SUBROUTINE INITIAL(MAP,MP,NP)                                     INITIAL.2
C                                                                       INITIAL.3
C     INITIALIZATION PRIOR TO SIMULATION                                INITIAL.4
C                                                                       INITIAL.5
C     EXTERNALS REFERENCED                                              INITIAL.6
C          ITL1                                                         INITIAL.7
C          ERK                                                          INITIAL.8
C     EXTERNALS CONTAINING REFERENCES                                   INITIAL.9
C          LSWAVE  (MAIN PROGRAM)                                       INITIAL.10
C                                                                       INITIAL.11
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),       //.2
     A     H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                  //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A     HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,               /DATA/.3
     B       IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,               /DATA/.4
     C     I10B(4),I20B(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),         /DATA/.5
     D     JHIST(50),JOB(4),J10B(4),J20B(4),JWEIR(9),J1WEIR(9),         /DATA/.6
     E     J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR      /DATA/.7
      COMMON /SEGM/ IK(300),I1(300),I2(300),JK(300),J1(300),J2(300),    /SEGM/.2
     A     MB(300),NB(300)                                              /SEGM/.3
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A     DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,     /SIMU/.3
     B     GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C     KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                  /SIMU/.5
      COMMON /HIST/ ES(2500),US(2500),VS(2500),Q(500),SS(500)           /HIST/.2
      COMMON /DIMS/ MAXES,MAXES1,MAXES2,IDIMD,IDIME,IDIMEH,IDIMOB,      /DIMS/.2
     A     IDIMTS,IDIMWE                                                /DIMS/.3
      DIMENSION MAP(MP,NP)                                              INITIAL.18
      DIMENSION D0(2000)                                                INITIAL.19
      EQUIVALENCE (D0,C)                                                INITIAL.20
      LOGICAL START                                                     INITIAL.21
      DATA PI/3.141592654/                                              INITIAL.22
C                                                                       INITIAL.23
C     DETERMINE DEPTH                                                   INITIAL.24
C                                                                       INITIAL.25
      IF (Z0 .EQ. 0.) GOTO 1920                                         INITIAL.26
      KNM=N*M                                                           INITIAL.27
      DO 1610 K=1,KNM                                                   INITIAL.28
```

```
 1010 IF (D(K) .NE. 0.) D(K)=ZO-D(K)                                    INITIAL.29
 1020 CONTINUE                                                          INITIAL.30
C                                                                       INITIAL.31
C     DETERMINE COMPUTATIONAL SEGMENTS AND                              INITIAL.32
C     SET UP MAP FOR COMPUTATIONAL POINTS                               INITIAL.33
C                                                                       INITIAL.34
      DO 1110 J=1,N                                                     INITIAL.35
      DO 1110 I=1,M                                                     INITIAL.36
 1110 MAP(I,J)=0                                                        INITIAL.37
      IJ=MS=NS=0                                                        INITIAL.38
      DO 1260 I=1,M                                                     INITIAL.39
      START=.FALSE.                                                     INITIAL.40
      DO 1250 J=1,N                                                     INITIAL.41
      IJ=IJ+1                                                           INITIAL.42
      IF (START) GOTO 1210                                              INITIAL.43
      IF (D(IJ)) 1240,1250,1200                                         INITIAL.44
 1200 MS=MS+1                                                           INITIAL.45
      IF ((I.EQ.1).OR.(I.EQ.M).OR.(J.EQ.1).OR.(J.EQ.N)) CALL ERR(21,1,  INITIAL.46
     ,  0,1H ,4,2,1HI,I,2,1HJ,J,2,1HM,M,2,1HN,N)                        INITIAL.47
      MAP(I,J)=1                                                        INITIAL.48
      MAP(I-1,J-1)=1                                                    INITIAL.49
      MAP(I-1,J)=1                                                      INITIAL.50
      MAP(I,J-1)=1                                                      INITIAL.51
      MAP(I+1,J-1)=1                                                    INITIAL.52
      MAP(I+1,J)=1                                                      INITIAL.53
      IK(MS)=I                                                          INITIAL.54
      J1(MS)=J                                                          INITIAL.55
      START=.TRUE.                                                      INITIAL.56
      GOTO 1250                                                         INITIAL.57
 1210 IF (D(IJ) .GT. 0.) GOTO 1235                                      INITIAL.58
      J2(MS)=J-1                                                        INITIAL.59
      IF (J2(MS) .EQ. J1(MS)) MS=MS-1                                   INITIAL.60
      START=.FALSE.                                                     INITIAL.61
 1235 MAP(I-1,J)=1                                                      INITIAL.62
      MAP(I+1,J)=1                                                      INITIAL.63
 1240 MAP(I,J)=1                                                        INITIAL.64
 1250 CONTINUE                                                          INITIAL.65
      IF (START) CALL ERR(22,1,0,1H ,4,2,1HI,I,2,1HJ,J,2,2HIJ,IJ,       INITIAL.66
```

```
     ,   1,5HD(IJ),D(IJ))                                              INITIAL.67
1260 CONTINUE                                                         INITIAL.68
     MAXCPM=1                                                         INITIAL.69
     DO 1295 I=1,M                                                    INITIAL.70
     DO 1295 J=1,N                                                    INITIAL.71
     IF (MAP(I,J) .EQ. 0) GOTO 1295                                   INITIAL.72
     MAXCPM=MAXCPM+1                                                  INITIAL.73
     MAP(I,J)=MAXCPM                                                  INITIAL.74
1295 CONTINUE                                                         INITIAL.75
     IF (MAXCPM .GT. IDIMEH) CALL ERR(23,1,0,1H ,2,2,6HMAXCPM,MAXCPM, INITIAL.76
     ,   2,6HIDIMEH,IDIMEH)                                           INITIAL.77
     MAXCP=MAXCPM+1                                                   INITIAL.78
     MAXCPN=MAXCPM+N                                                  INITIAL.79
     IF (MAXCPN .GT. IDIME) CALL ERR(24,1,0,1H ,2,2,6HMAXCPN,MAXCPN,  INITIAL.80
     ,   2,5HIDIME,IDIME)                                             INITIAL.81
     DO 1360 I=1,MAXCPM                                               INITIAL.82
1360 DD(I)=0.                                                         INITIAL.83
     DO 1470 J=1,N                                                    INITIAL.84
     START=.FALSE.                                                    INITIAL.85
     IJ=J                                                             INITIAL.86
     DO 1460 I=1,M                                                    INITIAL.87
     IF (START) GOTO 1425                                             INITIAL.88
     IF (D(IJ)) 1450,1455,1400                                        INITIAL.89
1400 NS=NS+1                                                          INITIAL.90
     IF ((I.EQ.1).OR.(I.EQ.M).OR.(J.EQ.1).OR.(J.EQ.N)) CALL ERR(25,1, INITIAL.91
     ,  0,1H ,4,2,1HI,I,2,1HJ,J,2,1HM,M,2,1HN,N)                      INITIAL.92
     JK(NS)=J                                                         INITIAL.93
     I1(NS)=I                                                         INITIAL.94
     START=.TRUE.                                                     INITIAL.95
     GOTO 1450                                                        INITIAL.96
1425 IF (D(IJ) .GT. 0.) GOTO 1450                                     INITIAL.97
     I2(NS)=I-1                                                       INITIAL.98
     IF (I2(NS) .EQ. I1(NS)) NS=NS-1                                  INITIAL.99
     START=.FALSE.                                                    INITIAL.100
     IF (D(IJ) .EQ. 0.) GOTO 1455                                     INITIAL.101
1450 DD(MAP(I,J))=D(IJ)                                               INITIAL.102
1455 IJ=IJ+N                                                          INITIAL.103
1460 CONTINUE                                                         INITIAL.104
```

```
      IF (START) CALL ERR(26,1,0,1H ,4,2,1HI,I,2,1HJ,J,2,2HIJ,IJ,        INITIAL.105
     ,   1,5HD(IJ),D(IJ))                                                 INITIAL.106
 1470 CONTINUE                                                            INITIAL.107
      DO 1500 K=MAXCPM,MAXCPN                                             INITIAL.108
 1500 D(K)=0.                                                             INITIAL.109
      DO 1505 K=1,MAXCPM                                                  INITIAL.110
 1505 D(K)=D0(K)*DCONV                                                    INITIAL.111
      DO 1545 K=1,MES                                                     INITIAL.112
      JES(K)=MAP(IHIST(K),JHIST(K))                                       INITIAL.113
 1545 CONTINUE                                                            INITIAL.114
C                                                                         INITIAL.115
C     DETERMINE TYPE OF BOUNDARY FOR ENDS OF EACH COMP SEGMENT            INITIAL.116
C                                                                         INITIAL.117
      DO 1575 I=1,MS                                                      INITIAL.118
 1575 MB(I)=11                                                            INITIAL.119
      DO 1580 I=1,NS                                                      INITIAL.120
 1580 NB(I)=11                                                            INITIAL.121
      DO 1730 K=1,MS                                                      INITIAL.122
      I=IK(K)                                                             INITIAL.123
      J1K=J1(K)                                                           INITIAL.124
      J2K=J2(K)                                                           INITIAL.125
      IF ((J1K .EQ. 1) .OR. (J2K .EQ. N)) CALL ERR(27,1,0,1H ,6,          INITIAL.126
     ,   2,1HN,N,2,3HJ1K,J1K,2,3HJ2K,J2K,2,2HMS,MS,2,2HNS,NS,2,1HI,I)     INITIAL.127
      IF (MAP(I,J1K-1) .EQ. 0) MAP(I,J1K-1)=MAXCP                         INITIAL.128
      IF (MAP(I,J2K+1) .EQ. 0) MAP(I,J2K+1)=MAXCP                         INITIAL.129
      IF (NWEIR .EQ. 0) GOTO 1680                                         INITIAL.130
      DO 1675 K9=1,NWEIR                                                  INITIAL.131
      IF ((I1WEIR(K9) .GT. I) .OR. (I2WEIR(K9) .LT. I)) GOTO 1675         INITIAL.132
      IF (JWEIR(K9) .EQ. J2K) GOTO 1665                                   INITIAL.133
      IF (JWEIR(K9) .NE. J1K) GOTO  1675                                  INITIAL.134
      MB(K)=MB(K)+40                                                      INITIAL.135
      HWEJ1=HWEIRJ(K9)                                                    INITIAL.136
      GOTO 1675                                                           INITIAL.137
 1665 MB(K)=MB(K)+4                                                       INITIAL.138
      HWEJ2=HWEIRJ(K9)                                                    INITIAL.139
 1675 CONTINUE                                                            INITIAL.140
 1680 IF (NOB .EQ. 0) GOTO 1730                                           INITIAL.141
      DO 1725 K9=1,NOB                                                    INITIAL.142
```

```
      IF ((I1OB(K9) .GT. I) .OR. (I2OB(K9) .LT. I)) GOTO 1725          INITIAL.143
      IF (JOB(K9) .EQ. J2K) GOTO 1720                                  INITIAL.144
      IF (JOB(K9) .NE. J1K) GOTO 1725                                  INITIAL.145
      MB(K)=MB(K)-10                                                   INITIAL.146
      GOTO 1725                                                        INITIAL.147
 1720 MB(K)=MB(K)-1                                                    INITIAL.148
 1725 CONTINUE                                                         INITIAL.149
 1730 CONTINUE                                                         INITIAL.150
C                                                                      INITIAL.151
      DO 1885 K=1,NS                                                   INITIAL.152
      J=JK(K)                                                          INITIAL.153
      I1K=I1(K)                                                        INITIAL.154
      I2K=I2(K)                                                        INITIAL.155
      IF ((I1K .EQ. 1) .OR. (I2K .EQ. M)) CALL ERR(28,1,0,1H ,6,       INITIAL.156
     ,   2,1HN,N,2,3HI1K,I1K,2,3HI2K,I2K,2,2HMS,MS,2,2HNS,NS,2,1HI,I)  INITIAL.157
      IF (MAP(I1K-1,J) .EQ. 0) MAP(I1K-1,J)=MAXCP                      INITIAL.158
      IF (MAP(I2K+1,J) .EQ. 0) MAP(I2K+1,J)=MAXCP                      INITIAL.159
      IF (MWEIR .EQ. 0) GOTO 1835                                      INITIAL.160
      DO 1830 K9=1,MWEIR                                               INITIAL.161
      IF ((J1WEIR(K9) .GT. J) .OR. (J2WEIR(K9) .LT. J)) GOTO 1830      INITIAL.162
      IF (IWEIR(K9) .EQ. I2K) GOTO 1820                                INITIAL.163
      IF (IWEIR(K9) .NE. I1K) GOTO 1830                                INITIAL.164
      NB(K)=NB(K)+40                                                   INITIAL.165
      HWEI1=HWEIR1(K9)                                                 INITIAL.166
      GOTO 1830                                                        INITIAL.167
 1820 NB(K)=NB(K)+4                                                    INITIAL.168
      HWEI2=HWEIR1(K9)                                                 INITIAL.169
 1830 CONTINUE                                                         INITIAL.170
 1835 IF (MOB .EQ. 0) GOTO 1885                                        INITIAL.171
      DO 1880 K9=1,MOB                                                 INITIAL.172
      IF ((J1OB(K9) .GT. J) .OR. (J2OB(K9) .LT. J)) GOTO 1880          INITIAL.173
      IF (IOB(K9) .EQ. I2K) GOTO 1875                                  INITIAL.174
      IF (IOB(K9) .NE. I1K) GOTO 1880                                  INITIAL.175
      NB(K)=NB(K)-10                                                   INITIAL.176
      GOTO 1880                                                        INITIAL.177
 1875 NB(K)=NB(K)-1                                                    INITIAL.178
 1880 CONTINUE                                                         INITIAL.179
 1885 CONTINUE                                                         INITIAL.180
```

```
      IF (MOB .EQ. 0) GOTO 1935                                         INITIAL.181
      DO 1930 K=1,MOB                                                   INITIAL.182
      I=IOB(K)                                                          INITIAL.183
      IF (J1OB(K) .EQ. 1) GOTO 1920                                     INITIAL.184
      IF (MAP(I,J1OB(K)-1) .EQ. 0) MAP(I,J1OB(K)-1)=MAXCP               INITIAL.185
 1920 IF (J2OB(K) .EQ. N) GOTO 1930                                     INITIAL.186
      IF (MAP(I,J2OB(K)+1) .EQ. 0) MAP(I,J2OB(K)+1)=MAXCP               INITIAL.187
 1930 CONTINUE                                                          INITIAL.188
 1935 IF (NOB .EQ. 0) GOTO 1975                                         INITIAL.189
      DO 1970 K=1,NOB                                                   INITIAL.190
      J=JOB(K)                                                          INITIAL.191
      IF (I1OB(K) .EQ. 1) GOTO 1960                                     INITIAL.192
      IF (MAP(I1OB(K)-1,J) .EQ. 0) MAP(I1OB(K)-1,J)=MAXCP               INITIAL.193
 1960 IF (I2OB(K) .EQ. M) GOTO 1970                                     INITIAL.194
      IF (MAP(I2OB(K)+1,J) .EQ. 0) MAP(I2OB(K)+1,J)=MAXCP               INITIAL.195
 1970 CONTINUE                                                          INITIAL.196
C                                                                       INITIAL.197
C     INITIAL VALUES                                                    INITIAL.198
C                                                                       INITIAL.199
 1975 CONTINUE                                                          INITIAL.200
      KES=KT=0                                                          INITIAL.201
      IES=KS=1                                                          INITIAL.202
      QT=0.                                                             INITIAL.203
      T=TZERO                                                           INITIAL.204
      DO 2010 K=1,MAXCPN                                                INITIAL.205
 2010 E(K)=0.                                                           INITIAL.206
      DO 2020 K=1,MAXCPM                                                INITIAL.207
 2020 IF (D(K) .LT. 0.) E(K)=-D(K)                                      INITIAL.208
      DO 2030 K=1,MAXCPM                                                INITIAL.209
 2030 EH(K)=0.                                                          INITIAL.210
      DO 2040 I=1,NES                                                   INITIAL.211
      SS(I)=0.                                                          INITIAL.212
 2040 Q(I)=0.                                                           INITIAL.213
      DO 2100 K=1,MAXCPN                                                INITIAL.214
      H2U(K)=0.                                                         INITIAL.215
      H2V(K)=0.                                                         INITIAL.216
      U(K)=0.                                                           INITIAL.217
      UN(K)=0.                                                          INITIAL.218
```

```
      V(K)=0.                                                           INITIAL.219
      VN(K)=0.                                                          INITIAL.220
      H(K)=D(K)+E(K)                                                    INITIAL.221
      HT(K)=H(K)                                                        INITIAL.222
 2100 CONTINUE                                                          INITIAL.223
      DTDX=DT/DX                                                        INITIAL.224
      DTDY=DT/DY                                                        INITIAL.225
      DXDT2=(DX*DX)/DT                                                  INITIAL.226
      DYDT2=(DY*DY)/DT                                                  INITIAL.227
      CDTDX=DTDX*IC                                                     INITIAL.228
      CDTDX5=CDTDX*0.5                                                  INITIAL.229
      CDTDY=DTDY*IC                                                     INITIAL.230
      CDTDY5=CDTDY*0.5                                                  INITIAL.231
      C3X=SQRT(G*0.5)*DTDX                                              INITIAL.232
      C3Y=SQRT(G*0.5)*DTDY                                              INITIAL.233
      IF (CD .LE. 0.) CD=3.33                                           INITIAL.234
      CD2=CD+CD                                                         INITIAL.235
      DTDX5=DTDX*0.5                                                    INITIAL.236
      DTDY5=DTDY*0.5                                                    INITIAL.237
      DX5=DX*0.5                                                        INITIAL.238
      DY5=DY*0.5                                                        INITIAL.239
      CALL ITL1(MWEIR,IWEIR,J1WEIR,J2WEIR,NWEIR,JWEIR,I1WEIR,I2WEIR,    INITIAL.240
     A      DX5)                                                        INITIAL.241
      CALL ITL1(NWEIR,JWEIR,I1WEIR,I2WEIR,MWEIR,IWEIR,J1WEIR,J2WEIR,    INITIAL.242
     A      DY5)                                                        INITIAL.243
      GDTDX=DTDX*G                                                      INITIAL.244
      GDTDY=DTDY*G                                                      INITIAL.245
      G5=0.5/G                                                          INITIAL.246
      T9=4./3.                                                          INITIAL.247
      DO 2200 J=1,MAXCPN                                                INITIAL.248
 2200 C(J)=AMAX1(D(J),0.1)**T9                                          INITIAL.249
      T9=1.486/CMANN                                                    INITIAL.250
      IF (G .LT. 20.) T9=1./CMANN                                       INITIAL.251
      IF (G .GT. 900.) T9=0.3201/CMANN                                  INITIAL.252
      CC=T9*T9*2./(G*DT)                                                INITIAL.253
      DO 2250 J=1,MAXCPN                                                INITIAL.254
 2250 C(J)=C(J)*CC                                                      INITIAL.255
      CTMK=CED+CED-1.                                                   INITIAL.256
```

```
DXDT2C=DXDT2*CED                                                INITIAL.257
DYDT2C=DYDT2*CED                                                INITIAL.258
RETURN                                                          INITIAL.259
END                                                             INITIAL.260
```

```
      SUBROUTINE INITFOR(MAP,MP,NP)                                     INITFOR.2
C                                                                       INITFOR.3
C     READ, PRINT, AND INITIALIZE DATA FOR FORCED LANDSLIDE             INITFOR.4
C                                                                       INITFOR.5
C     EXTERNALS REFERENCED                                              INITFOR.6
C          LANDSLD                                                      INITFOR.7
C          SLD1                                                         INITFOR.8
C          PRT1                                                         INITFOR.9
C          PRT5                                                         INITFOR.10
C          ERR                                                          INITFOR.11
C          SIN       (CDC LIBRARY)                                      INITFOR.12
C          COS       (CDC LIBRARY)                                      INITFOR.13
C     EXTERNALS CONTAINING REFERENCES                                   INITFOR.14
C          LSWAVE   (MAIN PROGRAM)                                      INITFOR.15
C                                                                       INITFOR.16
C                                                                       INITFOR.17
C ***********************************************************************VARFOR.2
C **  LIST OF INPUT VARIABLES FOR FORCED LANDSLIDE MODEL               **VARFOR.3
C ***********************************************************************VARFOR.4
C                                                                       VARFOR.5
C     TITLE(*)    SEVENTY CHARACTER SUBTITLE PRINTED WITH SLIDE DATA    VARFOR.6
C     NSL         NUMBER OF SLICES IN LANDSLIDE                         VARFOR.7
C     NXH         NUMBER OF HEIGHTS SPECIFIED FOR EACH SLICE            VARFOR.8
C     NTV         NUMBER OF VELOCITIES SPECIFIED FOR EACH SLICE         VARFOR.9
C     NSUB        NUMBER OF SUBDIVISIONS PER GRID ELEMENT IN THE        VARFOR.10
C                   X-DIRECTION FOR LANDSLIDE CALCULATIONS              VARFOR.11
C     NSUB        NUMBER OF SUBDIVISIONS PER GRID ELEMENT IN THE        VARFOR.12
C                   Y-DIRECTION FOR LANDSLIDE CALCULATIONS              VARFOR.13
C     THETA       FORWARD DIRECTION OF SLIDE (DEGREES COUNTERCLOCKWISE  VARFOR.14
C                   FROM X-AXIS)                                        VARFOR.15
C     WSL         WIDTH OF SLIDE (DISTANCE UNITS)                       VARFOR.16
C     XO          X-COORDINATE (GRID UNITS, NOT DISTANCE) OF RIGHT      VARFOR.17
C                   LEADING CORNER OF SLIDE (-RIGHT- IS DETERMINED AS IF VARFOR.18
C                   FACING IN THE DIRECTION OF FORWARD MOTION)          VARFOR.19
C     YO          Y-COORDINATE OF RIGHT LEADING CORNER OF SLIDE         VARFOR.20
C     CDP         PRESSURE DRAG COEFFICIENT (DEFAULTS TO 0.6 IF .LT. 0.) VARFOR.21
C     CDV         VISCOUS DRAG COEFFICIENT (DEFAULTS TO 0.008 IF .LT. 0.)VARFOR.22
C     TSLCNV      FACTOR TO CONVERT TSL(*) TO SECONDS                   VARFOR.23
```

```
C     VSLCNV      FACTOR TO CONVERT VSL(*) TO APPROPRIATE UNITS          VARFOR.24
C     XSL(*)      LOCATIONS WHERE SLIDE HEIGHT IS SPECIFIED (DISTANCES   VARFOR.25
C                   FROM LEADING EDGE)                                   VARFOR.26
C     HSL(I,J)    HEIGHT OF SLICE J AT DISTANCE XSL(I) FROM LEADING EDGE VARFOR.27
C     TSL(*)      TIMES AT WHICH SLIDE VELOCTIY IS SPECIFIED             VARFOR.28
C     VSL(I,J)    VELOCITY OF SLICE J AT TIME TSL(I)                     VARFOR.29
C                                                                        VARFOR.30
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),        //.2
     A       H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                 //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,          /DATA/.2
     A       HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,              /DATA/.3
     B         IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,              /DATA/.4
     C       I1OB(4),I2OB(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),        /DATA/.5
     D       JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),        /DATA/.6
     E       J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR     /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,       /SIMU/.2
     A       DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,    /SIMU/.3
     B       GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C       KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                 /SIMU/.5
      COMMON /HIST/ ES(2500),US(2500),VS(2500),Q(500),SS(500)            /HIST/.2
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,           /DSLD/.2
     A       WSL,DSL,X0,Y0,CDP,CDV,SGSL,AK,RHI,FRIC,TITLE(7),            /DSLD/.3
     B       XSL(101),TSL(101),HSL(101,10),VSL(101,10)                   /DSLD/.4
      COMMON /SLD1/ NSP(10),NKSP,IKSP(500),JKSP(500),XSN(10),VSN(10),    /SLD1/.2
     A       MA,NA,DXA,DYA,DXDXA,DYDXA,DXDYA,DYDYA,AMN,MNA,ITV,LSMAX(10), /SLD1/.3
     B       XA(2500)                                                    /SLD1/.4
      COMMON /SLD2/ XSP(500),LSP(500)                                    /SLD2/.2
      COMMON /SLD3/ HS(1350),HSN(1350),VSA(1350),ICV(1350),ICVN(1350)    /SLD3/.2
      COMMON /SDEF/ KDEF(10),ADEF(10),VDEF(10)                           /SDEF/.2
      DIMENSION MAP(MP,NP)                                               INITFOR.28
      REAL NUM(11,2)                                                     INITFOR.29
      LOGICAL NOSLID                                                     INITFOR.30
      DIMENSION Z(2000)                                                  INITFOR.31
      EQUIVALENCE (Z,ES)                                                 INITFOR.32
      INTEGER IDEF(20),KDEF,ADEF,VDEF                                    INITFOR.33
      REAL RDEF(20)                                                      INITFOR.34
      EQUIVALENCE (IDEF,NSL),(RDEF,NSL)                                  INITFOR.35
      LOGICAL NODEF                                                      INITFOR.36
```

```
      ALINEAR(X,X2,X1,Y2,Y1)=Y1+(X-X1)*(Y2-Y1)/(X2-X1)                  INITFOR.37
      DATA NUM/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H0,1H/,             INITFOR.38
     A         1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1H//             INITFOR.39
      DATA LSMAX/50,500,100,100,50,5*0/                                 INITFOR.40
      DATA ITV/2/                                                       INITFOR.41
      DATA NSP,NKSP/11*0/                                               INITFOR.42
      DATA XSN,VSN/20*0./                                               INITFOR.43
      DATA HS,HSN,VSA,ICV,ICVN/1350*0.,1350*0.,1350*0.,1350*0,1350*0/   INITFOR.44
      DATA XSL,TSL,HSL,VSL/2222*0./                                     INITFOR.45
      DATA KDEF/5,6,14,15,16,17,18,3*0/                                 INITFOR.46
      DATA ADEF/4HMSUB,4HNSUB,3HCDP,3HCDV,4HSGSL,2HAK,3HPHI/            INITFOR.47
      DATA VDEF/10,10,0.8,0.008,1.7,0.2,10./                            INITFOR.48
C                                                                       INITFOR.49
      READ 902,NSL,NXH,NTV,MSUB,NSUB                                    INITFOR.50
      READ 903,THETA,WSL,XO,YO,CDR,CDV                                  INITFOR.51
      READ 903,TSLCNV,VSLCNV                                            INITFOR.52
      READ 903,(XSL(I),I=1,NXH)                                         INITFOR.53
      DO 10 J=1,NSL                                                     INITFOR.54
   10 READ 903,(HSL(I,J),I=1,NXH)                                       INITFOR.55
      READ 903,(TSL(I),I=1,NTV)                                         INITFOR.56
      DO 20 J=1,NSL                                                     INITFOR.57
   20 READ 903,(VSL(I,J),I=1,NTV)                                       INITFOR.58
C                                                                       INITFOR.59
      PRINT 911                                                         INITFOR.60
      PRINT 912,TITLE                                                   INITFOR.61
      PRINT 913                                                         INITFOR.62
      CALL PRT1(12,3HNSL,NSL,3,29HNUMBER OF SLICES IN LANDSLIDE)        INITFOR.63
      CALL PRT1(12,3HNXH,NXH,5,42HNUMBER OF HEIGHTS SPECIFIED FOR EACH SINITFOR.64
     ,LICE)                                                             INITFOR.65
      CALL PRT1(12,3HNTV,NTV,5,45HNUMBER OF VELOCITIES SPECIFIED FOR EACINITFOR.66
     ,H SLICE)                                                          INITFOR.67
      CALL PRT1(2,4HMSUB,MSUB,8,77HNUMBER OF SUBDIVISIONS PER GRID IN THINITFOR.68
     ,E X-DIRECTION FOR LANDSLIDE CALCULATIONS)                         INITFOR.69
      CALL PRT1(2,4HNSUB,NSUB,8,77HNUMBER OF SUBDIVISIONS PER GRID IN THINITFOR.70
     ,E Y-DIRECTION FOR LANDSLIDE CALCULATIONS)                         INITFOR.71
      CALL PRT1(11,5HTHETA,THETA,6,74HANGLE FROM X-AXIS TO FORWARD DIRECINITFOR.72
     ,TION OF SLIDE (DEGREES COUNTERCLOCKWISE))                         INITFOR.73
      CALL PRT1(11,3HWSL,WSL,4,31HWIDTH OF SLIDE (DISTANCE UNITS))      INITFOR.74
```

```
      CALL PRT1(11,2HXO,XO,8,72HX-COORDINATE OF RIGHT LEADING CORNER OF INITFOR.75
     ,SLIDE (GRID UNITS, NOT DISTANCE))                                INITFOR.76
      CALL PRT1(11,2HYO,YO,6,58HY-COORDINATE OF RIGHT LEADING CORNER OF INITFOR.77
     ,SLIDE (GRID UNITS))                                              INITFOR.78
      CALL PRT1(11,3HCDP,CDP,3,25HPRESSURE DRAG COEFFICIENT)            INITFOR.79
      CALL PRT1(11,3HCDV,CDV,3,24HVISCOUS DRAG COEFFICIENT)             INITFOR.80
      CALL PRT1(11,6HTSLCNV,TSLCNV,4,35HFACTOR TO CONVERT TSL(*) TO SECOINITFOR.81
     ,NDS)                                                             INITFOR.82
      CALL PRT1(11,6HVSLCNV,VSLCNV,5,45HFACTOR TO CONVERT VSL(*) TO APPRINITFOR.83
     ,OPRIATE UNITS)                                                   INITFOR.84
C                                                                      INITFOR.85
C     SET DEFAULT VALUES FOR SLIDE DATA                                INITFOR.86
C                                                                      INITFOR.87
      VDEF(1)=LSMAX(5)                                                 INITFOR.88
      VDEF(2)=LSMAX(5)                                                 INITFOR.89
      NODEF=.TRUE.                                                     INITFOR.90
      NLP=24                                                           INITFOR.91
      DO 25 I=1,2                                                      INITFOR.92
      K=KDEF(I)                                                        INITFOR.93
      IF ((IDEF(K) .GT. 0) .AND. (IDEF(K) .LE. LSMAX(5))) GOTO 25      INITFOR.94
      IF (NODEF) PRINT 914                                             INITFOR.95
      NODEF=.FALSE.                                                    INITFOR.96
      NLP=NLP+1                                                        INITFOR.97
      IDEF(K)=VDEF(I)                                                  INITFOR.98
      CALL PRT1(12,ADEF(I),IDEF(K),0)                                  INITFOR.99
   25 CONTINUE                                                         INITFOR.100
      DO 30 I=3,4                                                      INITFOR.101
      K=KDEF(I)                                                        INITFOR.102
      IF (RDEF(K) .GE. 0.) GOTO 30                                     INITFOR.103
      IF (NODEF) PRINT 914                                             INITFOR.104
      NODEF=.FALSE.                                                    INITFOR.105
      NLP=NLP+1                                                        INITFOR.106
      IDEF(K)=VDEF(I)                                                  INITFOR.107
      CALL PRT1(11,ADEF(I),IDEF(K),0)                                  INITFOR.108
   30 CONTINUE                                                         INITFOR.109
      IF (.NOT. NODEF) NLP=NLP+8                                       INITFOR.110
C                                                                      INITFOR.111
      NLP=NLP+8+NXH                                                    INITFOR.112
```

```
      IF (NLP .LE. 63) GOTO 35                                          INITFOR.113
      NLP=8+NXH                                                         INITFOR.114
      PRINT 927                                                         INITFOR.115
   35 PRINT 921,(J,J=1,NSL)                                             INITFOR.116
      PRINT 926                                                         INITFOR.117
      DO 40 I=1,NXH                                                     INITFOR.118
   40 PRINT 924,I,XSL(I),(HSL(I,J),J=1,NSL)                             INITFOR.119
      NLP=NLP+8+NTV                                                     INITFOR.120
      IF (NLP .LE. 63) GOTO 42                                          INITFOR.121
      NLP=8+NTV                                                         INITFOR.122
      PRINT 927                                                         INITFOR.123
   42 PRINT 922                                                         INITFOR.124
      PRINT 923,(J,J=1,NSL)                                             INITFOR.125
      PRINT 926                                                         INITFOR.126
      DO 45 I=1,NTV                                                     INITFOR.127
   45 PRINT 924,I,TSL(I),(VSL(I,J),J=1,NSL)                             INITFOR.128
      IF ((TSLCNV .EQ. 1.) .AND. (VSLCNV .EQ. 1.)) GOTO 55              INITFOR.129
      DO 48 I=1,NTV                                                     INITFOR.130
      TSL(I)=TSL(I)*TSLCNV                                              INITFOR.131
      DO 48 J=1,NSL                                                     INITFOR.132
   48 VSL(I,J)=VSL(I,J)*VSLCNV                                          INITFOR.133
      NLP=NLP+8+NTV                                                     INITFOR.134
      IF (NLP .LE. 63) GOTO 49                                          INITFOR.135
      PRINT 927                                                         INITFOR.136
      NLP=8+NTV                                                         INITFOR.137
   49 CONTINUE                                                          INITFOR.138
      PRINT 925                                                         INITFOR.139
      PRINT 923,(J,J=1,NSL)                                             INITFOR.140
      PRINT 926                                                         INITFOR.141
      DO 52 I=1,NTV                                                     INITFOR.142
   52 PRINT 924,I,TSL(I),(VSL(I,J),J=1,NSL)                             INITFOR.143
   55 CONTINUE                                                          INITFOR.144
      IF (ABS(TZERO-TSL(1)) .GT. 0.1*DT) CALL ERR(53,1,3,29HTZERO MUST BINITFOR.145
     ,E EQUAL TO TSL(1),2,1,5HTZERO,TZERO,1,6HTSL(1),TSL(1))            INITFOR.146
      TZERO=TSL(1)                                                      INITFOR.147
C                                                                       INITFOR.148
C                                                                       INITFOR.149
      THETA=THETA*PI/180.                                               INITFOR.150
```

```
      COST=COS(THETA)                                                   INITFOR.151
      SINT=SIN(THETA)                                                   INITFOR.152
      DSL=WSL/NSL                                                       INITFOR.153
      XO=XO*DX                                                          INITFOR.154
      YO=YO*DY                                                          INITFOR.155
      CDV=CDV*0.5                                                       INITFOR.156
      CDP=CDP*0.5                                                       INITFOR.157
      MA=MSUB                                                           INITFOR.158
      NA=NSUB                                                           INITFOR.159
      DXA=DX/MA                                                         INITFOR.160
      DYA=DY/NA                                                         INITFOR.161
      DXDXA=DXA*COST                                                    INITFOR.162
      DYDXA=DXA*(-SINT)                                                 INITFOR.163
      DXDYA=DYA*SINT                                                    INITFOR.164
      DYDYA=DYA*COST                                                    INITFOR.165
      AMN=1./(MA*NA)                                                    INITFOR.166
      MNA=MA*NA                                                         INITFOR.167
C                                                                       INITFOR.168
      DO 70 K=1,NSL                                                     INITFOR.169
      DO 60 I=2,NTV                                                     INITFOR.170
   60 XSN(K)=XSN(K)+(VSL(I,K)+VSL(I-1,K))*(TSL(I)-TSL(I-1))             INITFOR.171
   70 XSN(K)=XSN(K)*0.5                                                 INITFOR.172
C                                                                       INITFOR.173
      MSL=LSMAX(2)/NSL                                                  INITFOR.174
      LSMAX(1)=MSL                                                      INITFOR.175
      CALL SLD1(MAP,M,N,XSP,LSP,MSL,NSL)                                INITFOR.176
      DO 77 K=1,NSL                                                     INITFOR.177
   77 XSN(K)=0.                                                         INITFOR.178
C                                                                       INITFOR.179
      IJZ=0                                                             INITFOR.180
      DO 80 I=1,M                                                       INITFOR.181
      DO 80 J=1,N                                                       INITFOR.182
      IJZ=IJZ+1                                                         INITFOR.183
      IJ=MAP(I,J)                                                       INITFOR.184
      Z(IJZ)=1H                                                         INITFOR.185
      IF (IJ .LE. 0) GOTO 80                                            INITFOR.186
      IF (IJ .GE. MAXCP) GOTO 80                                        INITFOR.187
      IF (D(IJ) .NE. 0.) Z(IJZ)=1H+                                     INITFOR.188
```

```
   80 CONTINUE                                                          INITFOR.189
      NOSLID=.TRUE.                                                     INITFOR.190
      DO 90 K=1,NSL                                                     INITFOR.191
      NSPK=NSP(K)                                                       INITFOR.192
      IF (NSPK .EQ. 0) GOTO 90                                          INITFOR.193
      NOSLID=.FALSE.                                                    INITFOR.194
      DO 85 I=1,NSPK                                                    INITFOR.195
      J=(K-1)*MSL+I                                                     INITFOR.196
      L=LSP(J)                                                          INITFOR.197
      IJZ=(IKSP(L)-1)*N+JKSP(L)                                         INITFOR.198
      ZIJZ=Z(IJZ)                                                       INITFOR.199
      JSYB=2                                                            INITFOR.200
      IF (XSP(J) .GT. -XSL(1)) JSYB=1                                   INITFOR.201
      ISYB=K                                                            INITFOR.202
      IF (ZIJZ .NE. 1H+) ISYB=11                                        INITFOR.203
   85 Z(IJZ)=NUM(ISYB,JSYB)                                             INITFOR.204
   90 CONTINUE                                                          INITFOR.205
      NLP=NLP+18+M                                                      INITFOR.206
      IF (NLP .LE. 63) GOTO 95                                          INITFOR.207
      NLP=18+M                                                          INITFOR.208
      PRINT 927                                                         INITFOR.209
   95 PRINT 930                                                         INITFOR.210
      CALL PRT5(Z,N,M)                                                  INITFOR.211
      IF (NOSLID) CALL ERR(51,1,6,54HLANDSLIDE TRACK CROSSES NO COMPUTATINITFOR.212
     ,IONAL GRID ELEMENTS)                                              INITFOR.213
C                                                                       INITFOR.214
C     COMPUTE INITIAL SUBMERGED DEPTH OF LANDSLIDE                      INITFOR.215
C                                                                       INITFOR.216
      CALL LANDSLD(MAP,M,N,XSP,LSP,MSL,NSL)                             INITFOR.217
      DO 100 K=1,NKSP                                                   INITFOR.218
      I=IKSP(K)                                                         INITFOR.219
      J=JKSP(K)                                                         INITFOR.220
      IJ=MAP(I,J)                                                       INITFOR.221
      IF (D(IJ) .LT. 0.) E(IJ)=-D(IJ)                                   INITFOR.222
      IF (D(IJ) .GE. 0.) E(IJ)=0.                                       INITFOR.223
      H(IJ)=HT(IJ)=D(IJ)+E(IJ)                                          INITFOR.224
  100 CONTINUE                                                          INITFOR.225
      RETURN                                                            INITFOR.226
```

```
C                                                                       INITFOR.227
  902 FORMAT (16I5)                                                     INITFOR.228
  903 FORMAT (8F10.5)                                                   INITFOR.229
  911 FORMAT (1H1,5(1H-),18H FORCED LANDSLIDE ,109(1H-))                INITFOR.230
  912 FORMAT (3(/),1X8A10)                                              INITFOR.231
  913 FORMAT (3(/),11X8HVARIABLE,6X5HVALUE,11X11HDESCRIPTION            INITFOR.232
     A      /11X8(1H-),6X8(1H-),8X80(1H-)//)                            INITFOR.233
  914 FORMAT (3(/),44H DEFAULT VALUES SET FOR THE FOLLOWING INPUT ,     INITFOR.234
     A      9HVARIABLES//11X8HVARIABLE,6X5HVALUE/11X8(1H-),6X8(1H-)//)  INITFOR.235
  921 FORMAT (3(/),43H HEIGHT PROFILE (FUNCTION OF DISTANCE FROM ,      INITFOR.236
     A      13HLEADING EDGE)                                            INITFOR.237
     B      //11X1HI,6X8HDISTANCE,7X28HHEIGHT (HSL(I,J),J=1,NSL) --     INITFOR.238
     C      /8X7HI=1,NXH,4X6HXSL(I),4X10I8)                             INITFOR.239
  922 FORMAT (3(/),36H VELOCITY PROFILE (FUNCTION OF TIME))             INITFOR.240
  923 FORMAT (/11X1HI,8X4HTIME,9X30HVELOCITY (VSL(I,J),J=1,NSL) --      INITFOR.241
     A      /8X7HI=1,NTV,4X6HTSL(I),4X10I8)                             INITFOR.242
  924 FORMAT (10X12,4XF10.3,6X10(1XF7.3))                               INITFOR.243
  925 FORMAT (3(/),18H VELOCITY PROFILE ,                               INITFOR.244
     A      33HAFTER APPLYING CONVERSION FACTORS)                       INITFOR.245
  926 FORMAT (1H )                                                      INITFOR.246
  927 FORMAT (1H1)                                                      INITFOR.247
  930 FORMAT (3(/),41H PLOT OF SLIDE PATH ON COMPUTATIONAL GRID         INITFOR.248
     A      //5X53H1, 2, ...  -  GRID ELEMENT CROSSED BY A SINGLE SLICE ,INITFOR.249
     B      24H (DIGIT IS SLICE NUMBER)                                 INITFOR.250
     C      /5X52HA, B, ...  -  GRID ELEMENT INITIALLY UNDER A SINGLE ,  INITFOR.251
     D      54HSLICE  (LETTERS CORRESPOND TO SLICE NUMBER, A=1, ETC.)   INITFOR.252
     E      /13X51H/  -  GRID ELEMENT CROSSED BY, OR INITIALLY UNDER, ,  INITFOR.253
     F      18HTWO OR MORE SLICES                                       INITFOR.254
     G      /13X45H+  -  COMPUTATIONAL ELEMENT NOT IN SLIDE PATH        INITFOR.255
     H      /7X23H(BLANK)  -  DRY ELEMENT//)                            INITFOR.256
      END                                                               INITFOR.257
```

```
      SUBROUTINE INITNAT(MAP,MP,NF)                                     INITNAT.2
C                                                                       INITNAT.3
C     READ, PRINT, AND INITIALIZE DATA FOR NATURAL LANDSLIDE            INITNAT.4
C                                                                       INITNAT.5
C     EXTERNALS REFERENCED                                              INITNAT.6
C          LANDSLD                                                      INITNAT.7
C          SLD1                                                         INITNAT.8
C          SLD2                                                         INITNAT.9
C          SLD3                                                         INITNAT.10
C          SLD4                                                         INITNAT.11
C          PRT1                                                         INITNAT.12
C          PRT5                                                         INITNAT.13
C          ERR                                                          INITNAT.14
C          SIN       (CDC LIBRARY)                                      INITNAT.15
C          COS       (CDC LIBRARY)                                      INITNAT.16
C     EXTERNALS CONTAINING REFERENCES                                   INITNAT.17
C          LSWAVE   (MAIN PROGRAM)                                      INITNAT.18
C                                                                       INITNAT.19
C                                                                       INITNAT.20
C ***********************************************************************VARNAT.2
C **  LIST OF INPUT VARIABLES FOR NATURAL LANDSLIDE MODEL              **VARNAT.3
C ***********************************************************************VARNAT.4
C                                                                       VARNAT.5
C     TITLE(*)    SEVENTY CHARACTER SUBTITLE PRINTED WITH SLIDE DATA    VARNAT.6
C     NSL         (UNUSED)                                              VARNAT.7
C     NXH         NUMBER OF HEIGHTS SPECIFIED FOR EACH SLICE            VARNAT.8
C     NTV         (UNUSED)                                              VARNAT.9
C     MSUB        NUMBER OF SUBDIVISIONS PER GRID ELEMENT IN THE        VARNAT.10
C                   X-DIRECTION FOR LANDSLIDE CALCULATIONS              VARNAT.11
C     NSUB        NUMBER OF SUBDIVISIONS PER GRID ELEMENT IN THE        VARNAT.12
C                   Y-DIRECTION FOR LANDSLIDE CALCULATIONS              VARNAT.13
C     THETA       FORWARD DIRECTION OF SLIDE (DEGREES COUNTERCLOCKWISE  VARNAT.14
C                   FROM X-AXIS)                                        VARNAT.15
C     WSL         WIDTH OF SLIDE (DISTANCE UNITS)                       VARNAT.16
C     XO          X-COORDINATE (GRID UNITS, NOT DISTANCE) OF RIGHT      VARNAT.17
C                   LEADING CORNER OF SLIDE (-RIGHT- IS DETERMINED AS IF VARNAT.18
C                   FACING IN THE DIRECTION OF FORWARD MOTION)          VARNAT.19
C     YO          Y-COORDINATE OF RIGHT LEADING CORNER OF SLIDE         VARNAT.20
```

```
C     CDP          PRESSURE DRAG COEFFICIENT (DEFAULTS TO 0.8 IF .LT. 0.) VARNAT.21
C     CDV          VISCOUS DRAG COEFFICIENT (DEFAULTS TO 0.008 IF .LT. 0.)VARNAT.22
C     SGSL         SPECIFIC GRAVITY OF SLIDE (DEFAULTS TO 1.7 IF .LT. 0.) VARNAT.23
C     AK           ADDED MASS COEFFICIENT (DEFAULTS TO 0.2 IF .LT. 0.)    VARNAT.24
C     PHI          FRICTION ANGLE (DEGREES) (DEFAULTS TO 10. IF .LT. 0.)  VARNAT.25
C     XSL(*)       LOCATIONS WHERE SLIDE HEIGHT IS SPECIFIED (DISTANCES   VARNAT.26
C                    FROM LEADING EDGE)                                   VARNAT.27
C     HSL(I)       HEIGHT OF SLIDE AT DISTANCE XSL(I) FROM LEADING EDGE   VARNAT.28
C                    (HSL*WSL SHOULD REPRESENT THE CROSS-SECTIONAL AREA)  VARNAT.29
C                                                                         VARNAT.30
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),         //.2
     A       H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                  //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,           /DATA/.2
     A       HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,               /DATA/.3
     B         IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,               /DATA/.4
     C       I10B(4),I20B(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),         /DATA/.5
     D       JHIST(50),JOB(4),J10B(4),J20B(4),JWEIR(9),J1WEIR(9),         /DATA/.6
     E       J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR      /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,        /SIMU/.2
     A       DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,     /SIMU/.3
     B       GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C       KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                  /SIMU/.5
      COMMON /HIST/ ES(2500),US(2500),VS(2500),Q(500),SS(500)             /HIST/.2
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,            /DSLD/.2
     A       WSL,DSL,X0,Y0,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),             /DSLD/.3
     B       XSL(101),TSL(101),HSL(101,10),VSL(101,10)                    /DSLD/.4
      COMMON /SLD1/ NSP(10),NKSP,IKSP(500),JKSP(500),XSN(10),VSN(10),     /SLD1/.2
     A       MA,NA,DXA,DYA,DXDXA,DYDXA,DXDYA,DYDYA,AMN,MNA,ITV,LSMAX(10), /SLD1/.3
     B       XA(2500)                                                     /SLD1/.4
      COMMON /SLD2/ XSP(500),LSP(500)                                     /SLD2/.2
      COMMON /SLD3/ HS(1350),HSN(1350),VSA(1350),ICV(1350),ICVN(1350)     /SLD3/.2
      COMMON /SDEF/ KDEF(10),ADEF(10),VDEF(10)                            /SDEF/.2
      DIMENSION MAP(MP,NP)                                                INITNAT.31
      REAL NUM(11,2)                                                      INITNAT.32
      LOGICAL NOSLID                                                      INITNAT.33
      DIMENSION Z(2000)                                                   INITNAT.34
      EQUIVALENCE (Z,ES)                                                  INITNAT.35
      INTEGER IDEF(20),KDEF,ADEF,VDEF                                     INITNAT.36
```

```
      REAL RDEF(20)                                                     INITNAT.37
      EQUIVALENCE (IDEF,NSL),(RDEF,NSL)                                 INITNAT.38
      LOGICAL NODEF                                                     INITNAT.39
      ALINEAR(X,X2,X1,Y2,Y1)=Y1+(X-X1)*(Y2-Y1)/(X2-X1)                  INITNAT.40
      DATA NUM/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H0,1H/,             INITNAT.41
     A         1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1HX/             INITNAT.42
C                                                                       INITNAT.43
C...  DATA STATEMENTS INITIALIZING SLIDE VARIABLES ARE IN -INITFOR-     INITNAT.44
C                                                                       INITNAT.45
      READ 902,NSL,NXH,NTV,MSUB,NSUB                                    INITNAT.46
      NSL=1                                                             INITNAT.47
      NTV=0                                                             INITNAT.48
      READ 903,THETA,WSL,XO,YO,CDP,CDV                                  INITNAT.49
      READ 903,SGSL,AK,PHI                                              INITNAT.50
      READ 903,(XSL(I),I=1,NXH)                                         INITNAT.51
      DO 10 J=1,NSL                                                     INITNAT.52
   10 READ 903,(HSL(I,J),I=1,NXH)                                       INITNAT.53
C                                                                       INITNAT.54
      PRINT 911                                                         INITNAT.55
      PRINT 912,TITLE                                                   INITNAT.56
      PRINT 913                                                         INITNAT.57
      CALL PRT1(12,3HNXH,NXH,5,42HNUMBER OF HEIGHTS SPECIFIED FOR EACH SINITNAT.58
     ,LICE)                                                             INITNAT.59
      CALL PRT1(2,4HMSUB,MSUB,8,77HNUMBER OF SUBDIVISIONS PER GRID IN THINITNAT.60
     ,E X-DIRECTION FOR LANDSLIDE CALCULATIONS)                         INITNAT.61
      CALL PRT1(2,4HNSUB,NSUB,8,77HNUMBER OF SUBDIVISIONS PER GRID IN THINITNAT.62
     ,E Y-DIRECTION FOR LANDSLIDE CALCULATIONS)                         INITNAT.63
      CALL PRT1(11,5HTHETA,THETA,6,74HANGLE FROM X-AXIS TO FORWARD DIRECINITNAT.64
     ,TION OF SLIDE (DEGREES COUNTERCLOCKWISE))                         INITNAT.65
      CALL PRT1(11,3HWSL,WSL,4,31HWIDTH OF SLIDE (DISTANCE UNITS))      INITNAT.66
      CALL PRT1(11,2HXO,XO,8,72HX-COORDINATE OF RIGHT LEADING CORNER OF INITNAT.67
     ,SLIDE (GRID UNITS, NOT DISTANCE))                                 INITNAT.68
      CALL PRT1(11,2HYO,YO,5,46HY-COORDINATE OF RIGHT LEADING CORNER OF INITNAT.69
     ,SLIDE)                                                            INITNAT.70
      CALL PRT1(11,3HCDP,CDP,3,25HPRESSURE DRAG COEFFICIENT)            INITNAT.71
      CALL PRT1(11,3HCDV,CDV,3,24HVISCOUS DRAG COEFFICIENT)             INITNAT.72
      CALL PRT1(11,4HSGSL,SGSL,3,29HSPECIFIC GRAVITY OF THE SLIDE)      INITNAT.73
      CALL PRT1(11,2HAK,AK,3,22HADDED MASS COEFFICIENT)                 INITNAT.74
```

```
      CALL PRT1(11,3HPHI,PHI,3,24HFRICTION ANGLE (DEGREES))
C
C     SET DEFAULT VALUES FOR SLIDE DATA
C
      VDEF(1)=LSMAX(5)
      VDEF(2)=LSMAX(5)
      NODEF=.TRUE.
      NLP=23
      DO 20 I=1,2
      K=KDEF(I)
      IF ((IDEF(K) .GT. 0) .AND. (IDEF(K) .LE. LSMAX(5))) GOTO 20
      IF (NODEF) PRINT 914
      NODEF=.FALSE.
      NLP=NLP+1
      IDEF(K)=VDEF(1)
      CALL PRT1(12,ADEF(I),IDEF(K),0)
   20 CONTINUE
      DO 25 I=3,7
      K=KDEF(I)
      IF (RDEF(K) .GE. 0.) GOTO 25
      IF (NODEF) PRINT 914
      NODEF=.FALSE.
      NLP=NLP+1
      IDEF(K)=VDEF(I)
      CALL PRT1(11,ADEF(I),IDEF(K),0)
   25 CONTINUE
      IF (.NOT. NODEF) NLP=NLP+8
C
      NLP=NLP+8+NXH
      IF (NLP .LE. 63) GOTO 38
      NLP=8+NXH
      PRINT 927
   38 PRINT 921
      PRINT 926
      DO 40 I=1,NXH
   40 PRINT 924,I,XSL(I),(HSL(I,J),J=1,NSL)
C
C
```

```
      THETA=THETA*PI/180.                                               INITNAT.113
      COST=COS(THETA)                                                   INITNAT.114
      SINT=SIN(THETA)                                                   INITNAT.115
      DSL=WSL/NSL                                                       INITNAT.116
      XO=XO*DX                                                          INITNAT.117
      YO=YO*DY                                                          INITNAT.118
      CDP=CDP*0.5                                                       INITNAT.119
      CDV=CDV*0.5                                                       INITNAT.120
      MA=MSUB                                                           INITNAT.121
      NA=NSUB                                                           INITNAT.122
      DXA=DX/MA                                                         INITNAT.123
      DYA=DY/NA                                                         INITNAT.124
      DXDXA=DXA*COST                                                    INITNAT.125
      DYDXA=DXA*(-SINT)                                                 INITNAT.126
      DXDYA=DYA*SINT                                                    INITNAT.127
      DYDYA=DYA*COST                                                    INITNAT.128
      AMN=1./(MA*NA)                                                    INITNAT.129
      LSKEY=2                                                           INITNAT.130
      MNA=MA*NA                                                         INITNAT.131
C                                                                       INITNAT.132
      CALL SLD2                                                         INITNAT.133
      CALL SLD3(MAP,M,N)                                                INITNAT.134
      CALL SLD4                                                         INITNAT.135
C                                                                       INITNAT.136
      NLP=NLP+8+NTV                                                     INITNAT.137
      IF (NLP .LE. 63) GOTO 42                                          INITNAT.138
      NLP=8+NTV                                                         INITNAT.139
      PRINT 927                                                         INITNAT.140
   42 PRINT 922                                                         INITNAT.141
      PRINT 923                                                         INITNAT.142
      PRINT 926                                                         INITNAT.143
      DO 45 I=1,NTV                                                     INITNAT.144
   45 PRINT 924,I,TSL(I),(VSL(I,J),J=1,NSL)                             INITNAT.145
C                                                                       INITNAT.146
      DO 70 K=1,NSL                                                     INITNAT.147
      DO 60 I=2,NTV                                                     INITNAT.148
   60 XSN(K)=XSN(K)+(VSL(I,K)+VSL(I-1,K))*(TSL(I)-TSL(I-1))             INITNAT.149
   70 XSN(K)=XSN(K)*0.5                                                 INITNAT.150
```

```
C                                                                       INITNAT.151
      MSL=LSMAX(2)/NSL                                                  INITNAT.152
      LSMAX(1)=MSL                                                      INITNAT.153
      CALL SLD1(MAP,M,N,XSP,LSP,MSL,NSL)                                INITNAT.154
      DO 77 K=1,NSL                                                     INITNAT.155
   77 XSN(K)=0.                                                         INITNAT.156
C                                                                       INITNAT.157
      IJZ=0                                                             INITNAT.158
      DO 80 I=1,M                                                       INITNAT.159
      DO 80 J=1,N                                                       INITNAT.160
      IJZ=IJZ+1                                                         INITNAT.161
      IJ=MAF(I,J)                                                       INITNAT.162
      Z(IJZ)=1H                                                         INITNAT.163
      IF (IJ .LE. 0) GOTO 80                                            INITNAT.164
      IF (IJ .GE. MAXCP) GOTO 80                                        INITNAT.165
      IF (D(IJ) .NE. 0.) Z(IJZ)=1H+                                     INITNAT.166
   80 CONTINUE                                                          INITNAT.167
      NOSLID=.TRUE.                                                     INITNAT.168
      DO 90 K=1,NSL                                                     INITNAT.169
      NSPK=NSP(K)                                                       INITNAT.170
      IF (NSPK .EQ. 0) GOTO 90                                          INITNAT.171
      NOSLID=.FALSE.                                                    INITNAT.172
      DO 85 I=1,NSPK                                                    INITNAT.173
      J=(K-1)*MSL+I                                                     INITNAT.174
      L=LSP(J)                                                          INITNAT.175
      IJZ=(IKSP(L)-1)*N+JKSP(L)                                         INITNAT.176
      ZIJZ=Z(IJZ)                                                       INITNAT.177
      JSYB=2                                                            INITNAT.178
      IF (XSP(J) .GT. -XSL(1)) JSYB=1                                   INITNAT.179
C...  ISYB=K                                                            INITNAT.180
C...  IF (ZIJZ .NE. 1H+) ISYB=11                                        INITNAT.181
      ISYB=11                                                           INITNAT.182
   85 Z(IJZ)=NUM(ISYB,JSYB)                                             INITNAT.183
   90 CONTINUE                                                          INITNAT.184
      NLP=NLP+17+M                                                      INITNAT.185
      IF (NLP .LE. 63) GOTO 95                                          INITNAT.186
      NLP=17+M                                                          INITNAT.187
      PRINT 927                                                         INITNAT.188
```

```
   95 PRINT 930                                                         INITNAT.189
      CALL PRT5(Z,N,M)                                                  INITNAT.190
      IF (NOSLID) CALL ERR(55,1,6,54HLANDSLIDE TRACK CROSSES NO COMPUTATINITNAT.191
     ,IONAL GRID ELEMENTS)                                              INITNAT.192
C                                                                       INITNAT.193
C     COMPUTE INITIAL SUBMERGED DEPTH OF LANDSLIDE                      INITNAT.194
C                                                                       INITNAT.195
      CALL LANDSLD(MAP,M,N,XSP,LSP,MSL,NSL)                             INITNAT.196
      DO 100 K=1,NKSP                                                   INITNAT.197
      I=IKSP(K)                                                         INITNAT.198
      J=JKSP(K)                                                         INITNAT.199
      IJ=MAP(I,J)                                                       INITNAT.200
      IF (D(IJ) .LT. 0.) E(IJ)=-D(IJ)                                   INITNAT.201
      IF (D(IJ) .GE. 0.) E(IJ)=0.                                       INITNAT.202
      H(IJ)=HT(IJ)=D(IJ)+E(IJ)                                          INITNAT.203
  100 CONTINUE                                                          INITNAT.204
      RETURN                                                            INITNAT.205
C                                                                       INITNAT.206
  902 FORMAT (16I5)                                                     INITNAT.207
  903 FORMAT (8F10.5)                                                   INITNAT.208
  911 FORMAT (1H1,5(1H-),19H NATURAL LANDSLIDE ,108(1H-))               INITNAT.209
  912 FORMAT (3(/),1X8A10)                                              INITNAT.210
  913 FORMAT (3(/),11X8HVARIABLE,6X5HVALUE,11X11HDESCRIPTION            INITNAT.211
     A      /11X8(1H-),6X8(1H-),8X60(1H-)/)                             INITNAT.212
  914 FORMAT (3(/),44H DEFAULT VALUES SET FOR THE FOLLOWING INPUT ,     INITNAT.213
     A      9HVARIABLES//11X8HVARIABLE,6X5HVALUE/11X8(1H-),6X8(1H-)/)   INITNAT.214
  921 FORMAT (3(/),43H HEIGHT PROFILE (FUNCTION OF DISTANCE FROM ,      INITNAT.215
     A      13HLEADING EDGE)                                            INITNAT.216
     B      //11X1HI,6X8HDISTANCE,7X6HHEIGHT                            INITNAT.217
     C      /8X7HI=1,NXH,4X6HXSL(I),8X6HHSL(I)/)                        INITNAT.218
  922 FORMAT (3(/),45H COMPUTED VELOCITY PROFILE (FUNCTION OF TIME))    INITNAT.219
  923 FORMAT (/11X1HI,8X4HTIME,9X8HVELOCITY                             INITNAT.220
     A      /8X7HI=1,NTV,4X6HTSL(I),9X6HVSL(I)/)                        INITNAT.221
  924 FORMAT (10X12,4XF10.3,6X10(1XF7.3))                               INITNAT.222
  926 FORMAT (1H )                                                      INITNAT.223
  927 FORMAT (1H1)                                                      INITNAT.224
  930 FORMAT (3(/),41H PLOT OF SLIDE PATH ON COMPUTATIONAL GRID         INITNAT.225
     A      //13X39H/  -  GRID ELEMENT CROSSED BY THE SLIDE,            INITNAT.226
```

```
B      /13X44HX   -  GRID ELEMENT INITIALLY UNDER THE SLIDE,          INITNAT.227
G      /13X45H+   -  COMPUTATIONAL ELEMENT NOT IN SLIDE PATH          INITNAT.228
H      /7X23H(BLANK)  -  DRY ELEMENT//)                               INITNAT.229
 END                                                                  INITNAT.230
```

```
      SUBROUTINE INITFLT(MAP,MP,MF)                                     INITFLT.2
C                                                                       INITFLT.3
C     READ AND PRINT DATA FOR FAULT INDUCED BOTTOM MOTION               INITFLT.4
C     RAISE WATER SURFACE FOR A HEIGHT OF DFLT, IN ONE SHOT             INITFLT.5
C                                                                       INITFLT.6
C     EXTERNALS REFERENCED                                              INITFLT.7
C          PRT5                                                         INITFLT.8
C     EXTERNALS CONTAINING REFERENCES                                   INITFLT.9
C          LSWAVE  (MAIN PROGRAM)                                       INITFLT.10
C                                                                       INITFLT.11
C ***********************************************************************VARFLT.2
C **  LIST OF INPUT VARIABLES FOR BOTTOM FAULT MODEL                   **VARFLT.3
C ***********************************************************************VARFLT.4
C                                                                       VARFLT.5
C     TITLE(*)    SEVENTY CHARACTER SUBTITLE PRINTED WITH FAULT DATA    VARFLT.6
C     FI(*)       X-GRID COORDINATES OF TWO POINTS ON FAULT LINE        VARFLT.7
C                   DISPLACED REGION IS TO THE RIGHT OF THE LINE        VARFLT.8
C                   (-RIGHT- IS DETERMINED AS IF STANDING AT THE FIRST  VARFLT.9
C                    POINT, FACING THE SECOND)                          VARFLT.10
C     FJ(*)       Y-GRID COORDINATES FOR THE TWO POINTS DESCRIBING      VARFLT.11
C                   THE FAULT LINE                                      VARFLT.12
C     DFLT        MAGNITUDE OF DISPLACEMENT                             VARFLT.13
C                                                                       INITFLT.13
      COMMON C(1350),E(1350),EH(1350),H(1350),HT(1350),H2U(1350),       //.2
     A       H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                //.3
      COMMON /DATA/ D(2000),CD,CHANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A       HWEIR1(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,ZU,             /DATA/.3
     B         IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,             /DATA/.4
     C       I1OB(4),I2OB(4),IS,ISF,IWEIR(9),I1WEIR(9),I2WEIR(9),       /DATA/.5
     D       JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),       /DATA/.6
     E       J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR    /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTKK,C3X,C3Y,      /SIMU/.2
     A       DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,   /SIMU/.3
     B       GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50), /SIMU/.4
     C       KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                /SIMU/.5
      COMMON /HIST/ ES(2500),US(2500),VS(2500),Q(500),SS(500)           /HIST/.2
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,          /DSLD/.2
     A       WSL,DSL,XO,YO,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),           /DSLD/.3
```

```
B     XSL(101),TSL(101),HSL(101,10),VSL(101,10)                /DSLD/.4
      DIMENSION MAP(MP,NP)                                     INITFLT.19
      DIMENSION FI(2),FJ(2)                                    INITFLT.20
      DIMENSION Z(2000)                                        INITFLT.21
      EQUIVALENCE (Z,ES)                                       INITFLT.22
C                                                              INITFLT.23
      READ 902,(FI(K),FJ(K),K=1,2),DFLT                        INITFLT.24
      PRINT 911                                                INITFLT.25
      PRINT 912,TITLE                                          INITFLT.26
      PRINT 913                                                INITFLT.27
      PRINT 914,(K,FI(K),FJ(K),K=1,2)                          INITFLT.28
      PRINT 915,DFLT                                           INITFLT.29
      DFX=FI(2)-FI(1)                                          INITFLT.30
      DFY=FJ(2)-FJ(1)                                          INITFLT.31
      IJZ=0                                                    INITFLT.32
      IFLT=0                                                   INITFLT.33
      DO 10 I=1,M                                              INITFLT.34
      DO 10 J=1,N                                              INITFLT.35
      IJZ=IJZ+1                                                INITFLT.36
      Z(IJZ)=1H                                                INITFLT.37
      IJ=MAP(I,J)                                              INITFLT.38
      IF (IJ .LE. 0) GOTO 10                                   INITFLT.39
      IF (IJ .GE. MAXCP) GOTO 10                               INITFLT.40
      IF (D(IJ) .EQ. 0.) GOTO 10                               INITFLT.41
      YF=(FI(1)-I)*DFY+(J-FJ(1))*DFX                           INITFLT.42
      Z(IJZ)=1H+                                               INITFLT.43
      IF (YF .GT. 0.) GOTO 10                                  INITFLT.44
      Z(IJZ)=1HF                                               INITFLT.45
      IFLT=IFLT+1                                              INITFLT.46
      D(IJ)=D(IJ)-DFLT                                         INITFLT.47
      E(IJ)=E(IJ)+DFLT                                         INITFLT.48
   10 CONTINUE                                                 INITFLT.49
      PRINT 916                                                INITFLT.50
      CALL PRT5(Z,N,M)                                         INITFLT.51
C                                                              INITFLT.52
      LSKEY=1                                                  INITFLT.53
      RETURN                                                   INITFLT.54
C                                                              INITFLT.55
```

```
910 FORMAT (8F10.5)                                                    INITFLT.56
911 FORMAT (1H1,5(1H-),30H SECTION MOVEMENT DUE TO FAULT ,97(1H-))      INITFLT.57
912 FORMAT (3(/),8A10)                                                 INITFLT.58
913 FORMAT (3(/),44H POINTS ALONG FAULT LINE  (GRID COORDINATES)       INITFLT.59
   A      //9X1HK,8X5HFI(K),8X5HFJ(K)//)                               INITFLT.60
914 FORMAT (9X I1,2(8XF5.2))                                           INITFLT.61
915 FORMAT (3(/),41H MAGNITUDE OF DISPLACEMENT --      DFLT = ,F6.2)   INITFLT.62
916 FORMAT (3(/),28H EFFECT OF CELL INITIALIZATION                     INITFLT.63
   A      //9X22HF  -  DISPLACED REGION                                INITFLT.64
   B      /9X42H+  -  COMPUTATIONAL REGION (NOT DISPLACED)            INITFLT.65
   C      /3X35H(BLANK)  -  NON-COMPUTATIONAL REGION//)               INITFLT.66
    END                                                                INITFLT.67
```

```
      SUBROUTINE SIMULAT(MAP,MP,NP,ES,US,VS,G,NESP,MESP)                SIMULAT.2
C                                                                       SIMULAT.3
C     SIMULATE WAVE PROPAGATION AND DISSIPATION                         SIMULAT.4
C                                                                       SIMULAT.5
C     THIS IS THE CORE OF THE HYDRODYNAMIC SUBMODEL                     SIMULAT.6
C                                                                       SIMULAT.7
C     EXTERNALS REFERENCED                                              SIMULAT.8
C          FEEDBKU                                                      SIMULAT.9
C          FEEDBKV                                                      SIMULAT.10
C          LANDSLD                                                      SIMULAT.11
C          SHOTPRT                                                      SIMULAT.12
C          SQRT     (CDC LIBRARY)                                       SIMULAT.13
C     EXTERNALS CONTAINING REFERENCES                                   SIMULAT.14
C          LSWAVE  (MAIN PROGRAM)                                       SIMULAT.15
C                                                                       SIMULAT.16
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),       //.2
     A       H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A       HWEIR1(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,             /DATA/.3
     B         IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,             /DATA/.4
     C       I1OB(4),I2OB(4),IS,ISP,IWEIP(9),I1WEIR(9),I2WEIR(9),       /DATA/.5
     D       JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),       /DATA/.6
     E       J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR    /DATA/.7
      COMMON /SEGM/ IK(300),I1(300),I2(300),JK(300),J1(300),J2(300),    /SEGM/.2
     A       MB(300),NB(300)                                            /SEGM/.3
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A       DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,   /SIMU/.3
     B       GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50), /SIMU/.4
     C       KES,KS,KT,MAXCP,MAXCPN,MAXCPN,MS,NS,PI,QT,T                /SIMU/.5
      COMMON /DSLD/ NSL,MSL,MXH,NTV,MSUB,NSUB,THETA,COST,SINT,          /DSLD/.2
     A       WSL,DSL,XC,YO,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),           /DSLD/.3
     B       XSL(101),TSL(101),HSL(101,10),VSL(101,10)                  /DSLD/.4
      COMMON /SLD1/ MSP(10),MKSP,IKSP(500),JKSP(500),XSN(10),VSN(10),   /SLD1/.2
     A       MA,NA,DXA,DYA,DXDXA,DYDXA,DXDYA,DYDYA,AMN,MNA,ITV,LSMAX(10), /SLD1/.3
     B       XA(2500)                                                   /SLD1/.4
      COMMON /SLD2/ YSP(500),LSP(500)                                   /SLD2/.2
      DIMENSION MAP(MP,NP)                                              SIMULAT.24
      DIMENSION ES(NESP,MESP),US(NESP,MESP),VS(NESP,MESP),G(NESP)       SIMULAT.25
```

```
C                                                                       SIMULAT.26
      GOTO 4635                                                         SIMULAT.27
 2345 T=T+DT                                                            SIMULAT.28
      KT=KT+1                                                           SIMULAT.29
C                                                                       SIMULAT.3
C...  UPDATE TOTAL WATER DEPTH AND FRICTION COEFFICIENT                 SIMULAT.31
      DO 2355 K=1,MAXCPN                                                SIMULAT.32
 2355 HT(K)=D(K)+E(K)                                                   SIMULAT.33
      IF (ICC .EQ. 0) GOTO 2390                                         SIMULAT.34
      DO 2365 K=1,MAXCPN                                                SIMULAT.35
 2365 H(K)=HT(K)                                                        SIMULAT.36
      DO 2385 K=1,MAXCPN                                                SIMULAT.37
 2385 C(K)=CC*AMAX1(H(K),0.1)**1.333333                                 SIMULAT.38
 2390 CONTINUE                                                          SIMULAT.39
C                                                                       SIMULAT.40
C...  LANDSLIDE                                                         SIMULAT.41
      IF (LSKEY .EQ. 2) CALL LANDSLD(MAP,M,N,XSP,LSP,NSL,NSL)           SIMULAT.42
C                                                                       SIMULAT.43
C...  COMPUTE U                                                         SIMULAT.44
      IF (NS .EQ. 0) GOTO 2810                                          SIMULAT.45
      DO 2795 K=1,NS                                                    SIMULAT.46
      J=JK(K)                                                           SIMULAT.47
      I1K=I1(K)                                                         SIMULAT.48
      I2K=I2(K)-1                                                       SIMULAT.49
      I9=MOD(NB(K),10)                                                  SIMULAT.50
      IJ=MAP(I2(K),J)                                                   SIMULAT.51
      IMJ=MAP(I2K,J)                                                    SIMULAT.52
      IF (I9 .EQ. 5) GOTO 2555                                          SIMULAT.53
      IF ((HT(IMJ) .EQ. 0.) .OR. (HT(IJ) .EQ. 0.)) GOTO 2595            SIMULAT.54
      H2U(IJ)=(E(IMJ)-E(IJ))*SQRT(HT(IMJ)+HT(IJ))*C3X                   SIMULAT.55
      GOTO 2595                                                         SIMULAT.56
C                                                                       SIMULAT.57
 2555 T9=AMIN1(E(IJ)-HWEI2,HT(IJ))                                      SIMULAT.58
      IF (T9 .LE. 0.) GOTO 2585                                         SIMULAT.59
      T9=U(IMJ)*U(IMJ)*G5+T9                                            SIMULAT.60
      H2U(IJ)=SQRT(T9)*T9*CD2                                           SIMULAT.61
      UN(IJ)=H2U(IJ)/(H(IJ)+H(IJ))                                      SIMULAT.62
      Q(IES)=H2U(IJ)*DY5+Q(IES)                                         SIMULAT.63
```

```
      GOTO 2595                                                         SIMULAT.64
 2585 H2U(IJ)=UN(IJ)=0                                                  SIMULAT.65
C                                                                       SIMULAT.66
 2595 IJ=MAP(I1K-1,J)                                                   SIMULAT.67
      IPJ=MAP(I1K,J)                                                    SIMULAT.68
      IF (NB(K) .GT. 49) GOTO 2635                                      SIMULAT.69
      I9=MAP(I1K+1,J)                                                   SIMULAT.70
      IF ((HT(IPJ) .EQ. 0.) .OR. (HT(I9) .EQ. 0.)) GOTO 2675            SIMULAT.71
      H2U(IJ)=(E(I9)-E(IPJ))*SQRT(HT(I9)+HT(IPJ))*C3X                   SIMULAT.72
      GOTO 2675                                                         SIMULAT.73
C                                                                       SIMULAT.74
 2635 T9=AMIN1(E(IPJ)-HWEI1,HT(IPJ))                                    SIMULAT.75
      IF (T9 .LE. 0.) GOTO 2665                                         SIMULAT.76
      T9=U(IPJ)*U(IPJ)*G5+T9                                            SIMULAT.77
      H2U(IJ)=-SQRT(T9)*T9*CD2                                          SIMULAT.78
      UN(IJ)=H2U(IJ)/(H(IPJ)+H(IPJ))                                    SIMULAT.79
      Q(IES)=-H2U(IJ)*DY5+Q(IES)                                        SIMULAT.80
      GOTO 2675                                                         SIMULAT.81
 2665 H2U(IJ)=UN(IJ)=0.                                                 SIMULAT.82
C                                                                       SIMULAT.83
 2675 DO 2790 I=I1K,I2K                                                 SIMULAT.84
      IMJ=IJ                                                            SIMULAT.85
      IJ=IPJ                                                            SIMULAT.86
      IPJ=MAP(I+1,J)                                                    SIMULAT.87
      VM=(V(IJ-1)+V(IJ)+V(IPJ-1)+V(IPJ))*0.25                           SIMULAT.88
      UIJ=U(IJ)                                                         SIMULAT.89
      T9=AMIN1((U(IPJ)-U(IMJ))*CDTDX5+SQRT(UIJ*UIJ+VM*VM)               SIMULAT.90
     A      /(C(IJ)+C(IPJ)),1.)                                         SIMULAT.91
      UN(IJ)=(1.-T9)*UIJ-(U(IJ+1)-U(IJ-1))*VM*CDTDY5                    SIMULAT.92
     A      -(E(IPJ)-E(IJ))*GDTDX                                       SIMULAT.93
      H2U(IJ)=(H(IPJ)+H(IJ))*UN(IJ)                                     SIMULAT.94
      IF (H2U(IJ) .LT. 0.) GOTO 2760                                    SIMULAT.95
      IF (H2U(IJ)*DTDY5 .LE. HT(IJ)) GOTO 2790                          SIMULAT.96
      H2U(IJ)=HT(IJ)*DYDT2                                              SIMULAT.97
      GOTO 2785                                                         SIMULAT.98
 2760 IF (-H2U(IJ)*DTDY5 .LE. HT(IPJ)) GOTO 2790                        SIMULAT.99
      H2U(IJ)=-HT(IPJ)*DYDT2                                            SIMULAT.100
 2785 UN(IJ)=H2U(IJ)/(H(IPJ)+H(IJ))                                     SIMULAT.101
```

```
 2790 CONTINUE                                                          SIMULAT.102
 2795 CONTINUE                                                          SIMULAT.103
C                                                                       SIMULAT.104
C...  COMPUTE V                                                         SIMULAT.105
 2810 IF (MS .EQ. 0) GOTO 3305                                          SIMULAT.106
      DO 3125 K=1,MS                                                    SIMULAT.107
      I=IK(K)                                                           SIMULAT.108
      J1K=J1(K)                                                         SIMULAT.109
      J2K=J2(K)-1                                                       SIMULAT.110
      I9=MOD(MB(K),10)                                                  SIMULAT.111
      IJ=MAP(I,J2(K))                                                   SIMULAT.112
      IF (I9 .EQ. 5) GOTO 2865                                          SIMULAT.113
      IF ((HT(IJ-1) .EQ. 0.) .OR. (HT(IJ) .EQ. 0.)) GOTO 2905           SIMULAT.114
      H2V(IJ)=(E(IJ-1)-E(IJ))*SQRT(HT(IJ-1)+HT(IJ))*C3Y                 SIMULAT.115
      GOTO 2905                                                         SIMULAT.116
 2865 T9=AMIN1(E(IJ)-HWEJ2,HT(IJ))                                      SIMULAT.117
      IF (T9 .LE. 0.) GOTO 2895                                         SIMULAT.118
      T9=V(IJ-1)*V(IJ-1)*G5+T9                                          SIMULAT.119
      H2V(IJ)=SQRT(T9)*T9*CD2                                           SIMULAT.120
      VN(IJ)=H2V(IJ)/(H(IJ)+H(IJ))                                      SIMULAT.121
      Q(IES)=H2V(IJ)*DX5+Q(IES)                                         SIMULAT.122
      GOTO 2905                                                         SIMULAT.123
 2895 VN(IJ)=H2V(IJ)=0.                                                 SIMULAT.124
C                                                                       SIMULAT.125
 2905 J=J1K-1                                                           SIMULAT.126
      IJ=MAP(I,J)                                                       SIMULAT.127
      IJP=IJ+1                                                          SIMULAT.128
      IMJ=MAP(I-1,J)                                                    SIMULAT.129
      IPJ=MAP(I+1,J)                                                    SIMULAT.130
      IF (MB(K) .GT. 49) GOTO 2955                                      SIMULAT.131
      I9=IJP+1                                                          SIMULAT.132
      IF ((HT(I9) .EQ. 0.) .OR. (HT(IJP) .EQ. 0.)) GOTO 2995            SIMULAT.133
      H2V(IJ)=(E(I9)-E(IJP))*SQRT(HT(I9)+HT(IJP))*C3Y                   SIMULAT.134
      GOTO 2995                                                         SIMULAT.135
 2955 T9=AMIN1(E(IJP)-HWEJ1,HT(IJP))                                    SIMULAT.136
      IF (T9 .LE. 0.) GOTO 2965                                         SIMULAT.137
      T9=T9+V(IJP)*V(IJP)*G5                                            SIMULAT.138
      H2V(IJ)=-SQRT(T9)*T9*CD2                                          SIMULAT.139
```

```
      VN(IJ)=H2V(IJ)/(H(IJP)+H(IJ))                                     SIMULAT.140
      Q(IES)=-H2V(IJ)*DX5+Q(IES)                                        SIMULAT.141
      GOTO 2995                                                         SIMULAT.142
 2985 H2V(IJ)=VN(IJ)=0.                                                 SIMULAT.143
C                                                                       SIMULAT.144
 2995 DO 3120 J=J1K,J2K                                                 SIMULAT.145
      IJ=IJP                                                            SIMULAT.146
      IJP=IJ+1                                                          SIMULAT.147
      IMJ=IMJ+1                                                         SIMULAT.148
      IPJ=IPJ+1                                                         SIMULAT.149
      HIJ=H(IJ)                                                         SIMULAT.150
      UM=(U(IMJ)+U(IMJ+1)+U(IJ)+U(IJP))*0.25                            SIMULAT.151
      VIJ=V(IJ)                                                         SIMULAT.152
      T9=AMIN1((V(IJP)-V(IJ-1))*CDTDY5+SQRT(UM*UM+VIJ*VIJ)              SIMULAT.153
     A      /(C(IJ)+C(IJP)),1.)                                         SIMULAT.154
      VN(IJ)=(1.-T9)*VIJ-(V(IPJ)-V(IMJ))*UM*CDTDX5                      SIMULAT.155
     A      -(E(IJP)-E(IJ))*GDTDY                                       SIMULAT.156
      H2V(IJ)=(H(IJP)+HIJ)*VN(IJ)                                       SIMULAT.157
      IF (H2V(IJ) .LT. 0.) GOTO 3090                                    SIMULAT.158
      IF (H2V(IJ)*DTDX5 .LE. HT(IJ)) GOTO 3120                          SIMULAT.159
      H2V(IJ)=HT(IJ)*DXDT2                                              SIMULAT.160
      GOTO 3115                                                         SIMULAT.161
 3090 IF (-H2V(IJ)*DTDX5 .LE. HT(IJP)) GOTO 3120                        SIMULAT.162
      H2V(IJ)=-HT(IJP)*DXDT2                                            SIMULAT.163
 3115 VN(IJ)=H2V(IJ)/(H(IJP)+HIJ)                                       SIMULAT.164
 3120 CONTINUE                                                          SIMULAT.165
 3125 CONTINUE                                                          SIMULAT.166
C                                                                       SIMULAT.167
C...  COMPUTE E DUE TO U                                                SIMULAT.168
 3335 IF (NS .EQ. 0) GOTO 3770                                          SIMULAT.169
      DO 3620 K=1,NS                                                    SIMULAT.170
      J=JK(K)                                                           SIMULAT.171
      I1K=I1(K)                                                         SIMULAT.172
      I2K=I2(K)                                                         SIMULAT.173
      I9=MOD(NB(K),10)                                                  SIMULAT.174
      IF (I9 .EQ. 5) GOTO 3450                                          SIMULAT.175
      IJ=MAP(I2K,J)                                                     SIMULAT.176
      I2K=I2K-1                                                         SIMULAT.177
```

```
      IMJ=MAP(I2K,J)                                                    SIMULAT.178
      IF (I9 .EQ. 1) GOTO 3400                                          SIMULAT.179
      E(IJ)=F(IJ)+H2U(IJ)                                               SIMULAT.180
      IF (E(IJ) .LT. -D(IJ)) E(IJ)=-D(IJ)                               SIMULAT.181
      UN(IJ)=(H2U(IMJ)-H2U(IJ)*DXDT2)/(H(IMJ)+H(IJ))                    SIMULAT.182
      GOTO 3450                                                         SIMULAT.183
 3400 CONTINUE                                                          SIMULAT.184
      T6=H2U(IMJ)*CTMR-H2U(IJ)*DYDT2C                                   SIMULAT.185
      E(IJ)=(H2U(IMJ)-T6)*DTDY5+E(IJ)                                   SIMULAT.186
      IF (E(IJ)+D(IJ) .GE. 0.) GOTO 3440                                SIMULAT.187
      CALL FEEDBKU(IJ,IMJ,T6,H2U(IMJ))                                  SIMULAT.188
 3440 IF (H(IMJ)+H(IJ) .NE. 0.) GOTO 3445                               SIMULAT.189
      UN(IJ)=0.                                                         SIMULAT.190
      GOTO 3450                                                         SIMULAT.191
 3445 UN(IJ)=T6/(H(IMJ)+H(IJ))                                          SIMULAT.192
 3450 IF (NB(K) .GT. 49) GOTO 3565                                      SIMULAT.193
      IJ=MAP(I1K,J)                                                     SIMULAT.194
      IMJ=MAP(I1K-1,J)                                                  SIMULAT.195
      I1K=I1K+1                                                         SIMULAT.196
      IF (NB(K) .GT. 9) GOTO 3515                                       SIMULAT.197
      E(IJ)=E(IJ)+H2U(IMJ)                                              SIMULAT.198
      IF (E(IJ) .LT. -D(IJ)) E(IJ)=-D(IJ)                               SIMULAT.199
      UN(IMJ)=(H2U(IMJ)*DXDT2+H2U(IJ))/(H(MAP(I1K,J))+H(IJ))            SIMULAT.200
      GOTO 3570                                                         SIMULAT.201
 3515 CONTINUE                                                          SIMULAT.202
      T7=H2U(IJ)*CTMR+H2U(IMJ)*DYDT2C                                   SIMULAT.203
      E(IJ)=(T7-H2U(IJ))*DTDY5+E(IJ)                                    SIMULAT.204
      IF (E(IJ)+D(IJ) .GE. 0.) GOTO 3555                                SIMULAT.205
      CALL FEEDBKU(IJ,IMJ,H2U(IJ),T7)                                   SIMULAT.206
 3555 IF (H(MAP(I1K,J))+H(IJ) .NE. 0.) GOTO 3560                        SIMULAT.207
      UN(IMJ)=0.                                                        SIMULAT.208
      GOTO 3570                                                         SIMULAT.209
 3560 UN(IMJ)=T7/(H(MAP(I1K,J))+H(IJ))                                  SIMULAT.210
      GOTO 3570                                                         SIMULAT.211
 3565 IJ=MAP(I1K-1,J)                                                   SIMULAT.212
 3570 DO 3615 I=I1K,I2K                                                 SIMULAT.213
      IMJ=IJ                                                            SIMULAT.214
      IJ=MAP(I,J)                                                       SIMULAT.215
```

```
      E(IJ)=(H2U(IMJ)-H2U(IJ))*DTDY5+E(IJ)                              SIMULAT.216
      IF (E(IJ)+D(IJ) .GE. 0.) GOTO 3615                                SIMULAT.217
      CALL FEEDBKU(IJ,IMJ,H2U(IJ),H2U(IMJ))                             SIMULAT.218
 3615 CONTINUE                                                          SIMULAT.219
 3620 CONTINUE                                                          SIMULAT.220
C                                                                       SIMULAT.221
C...  COMPUTE E DUE TO V                                                SIMULAT.222
 3770 IF (MS .EQ. 0) GOTO 4635                                          SIMULAT.223
      DO 4080 K=1,MS                                                    SIMULAT.224
      I=IK(K)                                                           SIMULAT.225
      J1K=J1(K)                                                         SIMULAT.226
      J2K=J2(K)                                                         SIMULAT.227
      I9=MOD(MB(K),10)                                                  SIMULAT.228
      IF (I9 .EQ. 5) GOTO 3910                                          SIMULAT.229
      IJ=MAP(I,J2K)                                                     SIMULAT.230
      J2K=J2K-1                                                         SIMULAT.231
      IF (I9 .EQ. 0) GOTO 3875                                          SIMULAT.232
      T6=H2V(IJ-1)*CTMR-H2V(IJ)*DXDT2C                                  SIMULAT.233
      E(IJ)=(H2V(IJ-1)-T6)*DTDX5+E(IJ)                                  SIMULAT.234
      IF (E(IJ)+D(IJ) .GE. 0.) GOTO 3865                                SIMULAT.235
      IJM=IJ-1                                                          SIMULAT.236
      CALL FEEDBKV(IJ,IJM,T6,H2V(IJ-1))                                 SIMULAT.237
 3865 IF (H(IJ)+H(IJ-1) .NE. 0.) GOTO 3870                              SIMULAT.238
      VN(IJ)=0.                                                         SIMULAT.239
      GOTO 3910                                                         SIMULAT.240
 3870 VN(IJ)=T6/(H(IJ)+H(IJ-1))                                         SIMULAT.241
      GOTO 3910                                                         SIMULAT.242
 3875 E(IJ)=E(IJ)+H2V(IJ)                                               SIMULAT.243
      IF (E(IJ) .LT. -D(IJ)) E(IJ)=-D(IJ)                               SIMULAT.244
      VN(IJ)=(H2V(IJ-1)-H2V(IJ)*DXDT2)/(H(IJ-1)+H(IJ))                  SIMULAT.245
C                                                                       SIMULAT.246
 3910 IF (MB(K) .GT. 49) GOTO 4025                                      SIMULAT.247
      IJ=MAP(I,J1K)                                                     SIMULAT.248
      J1K=J1K+1                                                         SIMULAT.249
      IF (MB(K) .LT. 9) GOTO 3985                                       SIMULAT.250
      IJM=IJ-1                                                          SIMULAT.251
      T7=H2V(IJ)*CTMR+H2V(IJM)*DXDT2C                                   SIMULAT.252
      E(IJ)=(T7-H2V(IJ))*DTDX5+E(IJ)                                    SIMULAT.253
```

```
      IF (E(IJ)+D(IJ) .GE. 0.) GOTO 3975                                SIMULAT.254
      CALL FEEDBKV(IJ,IJM,H2V(IJ),T7)                                   SIMULAT.255
 3975 IF (H(IJ+1)+H(IJ) .NE. 0.) GOTO 3980                              SIMULAT.256
      VN(IJM)=0.                                                        SIMULAT.257
      GOTO 4030                                                         SIMULAT.258
 3980 VN(IJM)=T7/(H(IJ+1)+H(IJ))                                        SIMULAT.259
      GOTO 4030                                                         SIMULAT.260
 3985 E(IJ)=E(IJ)+H2V(IJ-1)                                             SIMULAT.261
      IF (E(IJ) .LT. -D(IJ)) E(IJ)=-D(IJ)                               SIMULAT.262
      VN(IJ-1)=(H2V(IJ-1)*DXDT2+H2V(IJ))/(H(MAP(I,J1K))+H(IJ))          SIMULAT.263
      GOTO 4030                                                         SIMULAT.264
 4025 IJ=MAP(I,J1K-1)                                                   SIMULAT.265
 4030 DO 4075 J=J1K,J2K                                                 SIMULAT.266
      IJ=IJ+1                                                           SIMULAT.267
      E(IJ)=(H2V(IJ-1)-H2V(IJ))*DTDX5+E(IJ)                             SIMULAT.268
      IF (E(IJ)+D(IJ) .GE. 0.) GOTO 4075                                SIMULAT.269
      IJM=IJ-1                                                          SIMULAT.270
      CALL FEEDBKV(IJ,IJM,H2V(IJ),H2V(IJ-1))                            SIMULAT.271
 4075 CONTINUE                                                          SIMULAT.272
 4080 CONTINUE                                                          SIMULAT.273
C                                                                       SIMULAT.274
 4635 DO 4640 K=1,MAXCPN                                                SIMULAT.275
      U(K)=UN(K)                                                        SIMULAT.276
      V(K)=VN(K)                                                        SIMULAT.277
 4640 CONTINUE                                                          SIMULAT.278
      IF (KT .NE. ITS(KS)) GOTO 4690                                    SIMULAT.279
      KS=KS+1                                                           SIMULAT.280
      IF (LSE .EQ. 0) GOTO 4670                                         SIMULAT.281
      CALL SHOTPRT(E,D,MAXCP,MAP,M,N,T,3,                               SIMULAT.282
     A     29HSNAPSHOT OF SURFACE ELEVATION,0)                          SIMULAT.283
 4670 IF (LSU .EQ. 0) GOTO 4690                                         SIMULAT.284
      CALL SHOTPRT(U,D,MAXCP,MAP,M,N,T,4,                               SIMULAT.285
     A     36HSNAPSHOT OF X-DIRECTION VELOCITY (U),0)                   SIMULAT.286
      CALL SHOTPRT(V,D,MAXCP,MAP,M,N,T,4,                               SIMULAT.287
     A     36HSNAPSHOT OF Y-DIRECTION VELOCITY (V),1)                   SIMULAT.288
 4690 IF (LENV .EQ. 0) GOTO 4715                                        SIMULAT.289
      DO 4700 J=1,MAXCPM                                                SIMULAT.290
 4700 IF (D(J)+L(J) .NE. 0.) EH(J)=AMAX1(EH(J),E(J))                    SIMULAT.291
```

```
4715 CONTINUE                                                          SIMULAT.292
     KES=KES+1                                                         SIMULAT.293
     IF (KES .LT. IS) GOTO 4755                                        SIMULAT.294
     KES=0                                                             SIMULAT.295
     IF (MES .EQ. 0) GOTO 4752                                         SIMULAT.296
     DO 4750 I=1,MES                                                   SIMULAT.297
     US(IES,I)=U(JES(I))                                               SIMULAT.298
     VS(IES,I)=V(JES(I))                                               SIMULAT.299
4750 ES(IES,I)=E(JES(I))                                               SIMULAT.300
4752 QT=Q(IES)+QT                                                      SIMULAT.301
     Q(IES)=Q(IES)/IS                                                  SIMULAT.302
     IES=IES+1                                                         SIMULAT.303
4755 IF (T .LT. TMAX) GOTO 2345                                        SIMULAT.304
     IES=IES-1                                                         SIMULAT.305
     RETURN                                                            SIMULAT.306
     END                                                               SIMULAT.307
```

```
      SUBROUTINE FEEDBKU(IJ,IMJ,T6,T7)                                  FEEDBKU.2
C                                                                       FEEDBKU.3
C     FEED BACK X-DIRECTION DISCHARGE TO A POINT WHEN THAT POINT IS     FEEDBKU.4
C       OVER-DRAINED                                                    FEEDBKU.5
C                                                                       FEEDBKU.6
C     EXTERNALS REFERENCED                                              FEEDBKU.7
C          (NONE)                                                       FEEDBKU.8
C     EXTERNALS CONTAINING REFERENCES                                   FEEDBKU.9
C          SIMULAT                                                      FEEDBKU.10
C                                                                       FEEDBKU.11
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),       //.2
     A       H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A       HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,             /DATA/.3
     B         IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,             /DATA/.4
     C       I1OB(4),I2OB(4),IS,ISR,IWEIR(9),I1WEIR(9),I2WEIR(9),       /DATA/.5
     D       JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),       /DATA/.6
     E       J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR    /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A       DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,   /SIMU/.3
     B       GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50), /SIMU/.4
     C       KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,FI,QT,T                /SIMU/.5
C                                                                       FEEDBKU.15
      T9=(D(IJ)+E(IJ))/((T6-T7)*DTDY5)+1.                               FEEDBKU.16
      IF (T9 .GE. 0.) GOTO 410                                          FEEDBKU.17
      IF (T9 .GE. -0.001) GOTO 400                                      FEEDBKU.18
C...  CALL ERR(31,0,3,25H-ERR- CALLED FROM FEEDBKU,9,                   FEEDBKU.19
C    ,   2,2HIJ,IJ,2,3HIMJ,IMJ,1,2HT6,T6,1,2HT7,T7,1,5HD(IJ),D(IJ),     FEEDBKU.20
C    ,   1,5HE(IJ),E(IJ),1,5HDTDY5,DTDY5,1,2HT9,T9,1,1HT,T)             FEEDBKU.21
  400 T9=0.                                                             FEEDBKU.22
  410 E(IJ)=-D(IJ)                                                      FEEDBKU.23
      UN(IJ)=UN(IJ)*T9                                                  FEEDBKU.24
      T6=T6*T9                                                          FEEDBKU.25
      IF (E(IMJ) .EQ. 0.) GOTO 460                                      FEEDBKU.26
      E(IMJ)=E(IMJ)+T7*(1.-T9)*DTDY5                                    FEEDBKU.27
      IF (E(IMJ) .LT. -D(IMJ)) E(IMJ)=-D(IMJ)                           FEEDBKU.28
  460 UN(IMJ)=UN(IMJ)*T9                                                FEEDBKU.29
      T7=T7*T9                                                          FEEDBKU.30
```

```
      RETURN                                                            FEEDBKU.31
      END                                                               FEEDBKU.32
```

```
      SUBROUTINE FEEDBKV(IJ,IJM,T6,T7)                                  FEEDBKV.2
C                                                                       FEEDBKV.3
C     FEED BACK Y-DIRECTION DISCHARGE TO A POINT WHEN THAT POINT IS     FEEDBKV.4
C       OVER-DRAINED                                                    FEEDBKV.5
C                                                                       FEEDBKV.6
C     EXTERNALS REFERENCED                                              FEEDBKV.7
C          (NONE)                                                       FEEDBKV.8
C     EXTERNALS CONTAINING REFERENCES                                   FEEDBKV.9
C          SIMULAT                                                      FEEDBKV.10
C                                                                       FEEDBKV.11
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),       //.2
     A       H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A       HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,ZO,             /DATA/.3
     B         IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,             /DATA/.4
     C       I1OB(4),I2OB(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),       /DATA/.5
     D       JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),       /DATA/.6
     E       J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR    /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A       DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,   /SIMU/.3
     B       GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50), /SIMU/.4
     C       KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                /SIMU/.5
C                                                                       FEEDBKV.15
      T9=(D(IJ)+E(IJ))/((T6-T7)*DTDX5)+1.                               FEEDBKV.16
      IF (T9 .GE. 0.) GOTO 510                                          FEEDBKV.17
      IF (T9 .GE. -0.001) GOTO 500                                      FEEDBKV.18
C...  CALL ERR(32,5,3,25H-ERR- CALLED FROM FEEDBKV,9,                   FEEDBKV.19
C    ,   2,2HIJ,IJ,2,3HIJM,IJM,1,2HT6,T6,1,2HT7,T7,1,5HD(IJ),D(IJ),     FEEDBKV.20
C    ,   1,5HE(IJ),E(IJ),1,5HDTDY5,DTDY5,1,2HT9,T9,1,1HT,T)             FEEDBKV.21
  500 T9=0.                                                             FEEDBKV.22
  510 E(IJ)=-D(IJ)                                                      FEEDBKV.23
      VN(IJ)=VN(IJ)*T9                                                  FEEDBKV.24
      T6=T6*T9                                                          FEEDBKV.25
      IF (E(IJM) .EQ. 0.) GOTO 560                                      FEEDBKV.26
      E(IJM)=E(IJM)+T7*(1.-T9)*DTDX5                                    FEEDBKV.27
      IF (E(IJM) .LT. -D(IJM)) E(IJM)=-D(IJM)                           FEEDBKV.28
  560 VN(IJM)=VN(IJM)*T9                                                FEEDBKV.29
      T7=T7*T9                                                          FEEDBKV.30
```

```
      RETURN                                                            FEEDBKV.31
      END                                                               FEEDBKV.32
```

```
      SUBROUTINE LANDSLD(MAP,MP,NP,XSP,LSP,MSLP,NSLP)                   LANDSLD.2
C                                                                       LANDSLD.3
C     SIMULATE LANDSLIDE MOTION FOR EACH TIME STEP                      LANDSLD.4
C                                                                       LANDSLD.5
C     EXTERNALS REFERENCED                                              LANDSLD.6
C          VSQRT                                                        LANDSLD.7
C          SQRT     (CDC LIBRARY)                                       LANDSLD.8
C     EXTERNALS CONTAINING REFERENCES                                   LANDSLD.9
C          INITFOR                                                      LANDSLD.10
C          INITNAT                                                      LANDSLD.11
C          SIMULAT                                                      LANDSLD.12
C                                                                       LANDSLD.13
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),       //.2
     A     H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                  //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A     HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,               /DATA/.3
     B       IC,ICC,IENV,IHIST(50),ILR(2,3),I0B(4),IPLTF,               /DATA/.4
     C     I10B(4),I20B(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),         /DATA/.5
     D     JHIST(50),J0B(4),J10B(4),J20B(4),JWEIR(9),J1WEIR(9),         /DATA/.6
     E     J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR      /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A     DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,     /SIMU/.3
     B     GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C     KES,KS,KT,MAXCP,MAXCBM,MAXCPN,MS,NS,PI,QT,T                  /SIMU/.5
      COMMON /HIST/ FS(2500),US(2500),VS(2500),Q(500),SS(500)           /HIST/.2
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,          /DSLD/.2
     A     WSL,DSL,X0,Y0,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),             /DSLD/.3
     B     XSL(101),TSL(101),HSL(101,10),VSL(101,10)                    /DSLD/.4
      COMMON /SLD1/ NSP(10),NKSP,IKSP(500),JKSP(500),XSN(10),VSN(10),   /SLD1/.2
     A     MA,NA,DXA,DYA,DXDXA,DYDXA,DXDYA,DYDYA,AMN,MNA,ITV,LSMAX(10), /SLD1/.3
     B     XA(2500)                                                     /SLD1/.4
      COMMON /SLD3/ HS(1350),HSN(1350),VSA(1350),ICV(1350),ICVN(1350)   /SLD3/.2
      DIMENSION MAP(MP,NP),XSP(MSLP,NSLP),LSP(MSLP,NSLP)                LANDSLD.21
      LOGICAL NOSLID                                                    LANDSLD.22
      ALINEAR(X,X2,X1,Y2,Y1)=Y1+(X-X1)*(Y2-Y1)/(X2-X1)                  LANDSLD.23
      DATA HSUM/0./                                                     LANDSLD.24
C                                                                       LANDSLD.25
      T1=T-DT                                                           LANDSLD.26
```

```
      IT1=ITV                                                           LANDSLD.27
   10 IF (T .LT. TSL(ITV)) GOTO 20                                      LANDSLD.28
      ITV=ITV+1                                                         LANDSLD.29
      IF (ITV .LE. NTV) GOTO 10                                         LANDSLD.30
      ITV=ITV-1                                                         LANDSLD.31
      LSKEY=1                                                           LANDSLD.32
C                                                                       LANDSLD.33
   20 IT2=ITV-1                                                         LANDSLD.34
      DO 40 K=1,NSL                                                     LANDSLD.35
      VSNK=ALINEAR(AMIN1(T,TSL(NTV)),TSL(ITV),TSL(IT2),                 LANDSLD.36
     A     VSL(ITV,K),VSL(IT2,K))                                       LANDSLD.37
      XSNK=VSN(K)*(AMIN1(T,TSL(IT1))-T1)+VSNK*(T-AMAX1(T1,TSL(IT2)))    LANDSLD.38
      VSN(K)=VSNK                                                       LANDSLD.39
      IF (IT2 .LT. IT1) GOTO 40                                         LANDSLD.40
      DO 30 ITK=IT1,IT2                                                 LANDSLD.41
   30 XSNK=XSNK+VSL(ITK,K)*(AMIN1(TSL(ITK+1),T)-AMAX1(TSL(ITK-1),T1))   LANDSLD.42
   40 XSN(K)=XSN(K)+0.5*XSNK                                            LANDSLD.43
C                                                                       LANDSLD.44
      DO 50 K=1,NKSP                                                    LANDSLD.45
      IJ=MAP(IKSP(K),JKSP(K))                                           LANDSLD.46
   50 HSN(IJ)=VSA(IJ)=0.                                                LANDSLD.47
C                                                                       LANDSLD.48
      NOSLID=.TRUE.                                                     LANDSLD.49
      DO 110 K=1,NSL                                                    LANDSLD.50
      NSPK=NSP(K)                                                       LANDSLD.51
      IF (NSPK .EQ. 0) GOTO 110                                         LANDSLD.52
      XSO=XO+XSN(K)*COST + (K-1)*DSL*(-SINT)                            LANDSLD.53
      YSO=YO+XSN(K)*SINT + (K-1)*DSL*COST                               LANDSLD.54
      DO 100 IX=1,NSPK                                                  LANDSLD.55
      IF (XSP(IX,K) .GT. XSN(K)) GOTO 110                               LANDSLD.56
      NOSLID=.FALSE.                                                    LANDSLD.57
      L=LSP(IX,K)                                                       LANDSLD.58
      I=IKSP(L)                                                         LANDSLD.59
      J=JKSP(L)                                                         LANDSLD.60
      NXA=0                                                             LANDSLD.61
      X=(I-0.5)*DX -0.5*DXA                                             LANDSLD.62
      Y=(J-0.5)*DY -0.5*DYA                                             LANDSLD.63
      XP=(X-XSO)*COST+(Y-YSO)*SINT                                      LANDSLD.64
```

```
      YP=(XSO-X)*SINT+(Y-YSO)*COST                                      LANDSLD.65
      DO 60 IA=1,MA                                                     LANDSLD.66
      X=XP+IA*DXDXA                                                     LANDSLD.67
      Y=YP+IA*DYDXA                                                     LANDSLD.68
      DO 60 JA=1,NA                                                     LANDSLD.69
      X=X+DXDYA                                                         LANDSLD.70
      Y=Y+DYDYA                                                         LANDSLD.71
      XAX=-X                                                            LANDSLD.72
      IF ((XAX .LT. XSL(1)) .OR. (XAX .GT. XSL(NXH))) GOTO 60           LANDSLD.73
      IF ((Y .GE. DSL) .OR. (Y .LT. 0.)) GOTO 60                        LANDSLD.74
      NXA=NXA+1                                                         LANDSLD.75
      XA(NXA)=XAX                                                       LANDSLD.76
   60 CONTINUE                                                          LANDSLD.77
      IF (NXA .EQ. 0) GOTO 100                                          LANDSLD.78
      CALL VSORT(XA,1,NXA)                                              LANDSLD.79
      IJ=MAP(I,J)                                                       LANDSLD.80
      IXSL=1                                                            LANDSLD.81
      HIJ=VIJ=0.                                                        LANDSLD.82
      DO 90 IXA=1,NXA                                                   LANDSLD.83
   70 IF (XA(IXA) .LT. XSL(IXSL)) GOTO 80                               LANDSLD.84
      IXSL=IXSL+1                                                       LANDSLD.85
      HSLOPE=(HSL(IXSL,K)-HSL(IXSL-1,K))/(XSL(IXSL)-XSL(IXSL-1))        LANDSLD.86
      IF (IXSL .LE. NXH) GOTO 70                                        LANDSLD.87
      IXSL=NXH                                                          LANDSLD.88
      HSLOPE=(HSL(IXSL,K)-HSL(IXSL-1,K))/(XSL(IXSL)-XSL(IXSL-1))        LANDSLD.89
   80 HIJA=HSL(IXSL,K)-HSLOPE*(XSL(IXSL)-XA(IXA))                       LANDSLD.90
      HIJ=HIJ+HIJA                                                      LANDSLD.91
   90 CONTINUE                                                          LANDSLD.92
      VSA(IJ)=VSA(IJ)+VSN(K)*NXA*AMN                                    LANDSLD.93
      ICVN(IJ)=ICVN(IJ)+NXA                                             LANDSLD.94
      HSN(IJ)=HSN(IJ)+HIJ*AMN                                           LANDSLD.95
  150 CONTINUE                                                          LANDSLD.96
  110 CONTINUE                                                          LANDSLD.97
      IF (NOSLID) RETURN                                                LANDSLD.98
C                                                                       LANDSLD.99
C     RAISE BOTH D AND E FOR HDIF DUE TO MASS INTRUDING INTO A GRID     LANDSLD.100
C                                                                       LANDSLD.101
      DO 400 K=1,NKSF                                                   LANDSLD.102
```

```
      I=IKSP(K)                                                         LANDSLD.103
      J=JKSP(K)                                                         LANDSLD.104
      IJ=MAP(I,J)                                                       LANDSLD.105
      HIJ=HSN(IJ)                                                       LANDSLD.106
      HDIF=HIJ-HS(IJ)                                                   LANDSLD.107
      NCV=MNA-ICV(IJ)                                                   LANDSLD.108
      NCVN=MAX0(ICVN(IJ)-ICV(IJ),0)                                     LANDSLD.109
      ICV(IJ)=MAX0(ICVN(IJ),ICV(IJ))                                    LANDSLD.110
      ICVN(IJ)=0                                                        LANDSLD.111
      HRCV=HT(IJ)                                                       LANDSLD.112
      IF (NCV .GT. 0) HRCV=(HRCV*NCVN)/NCV                              LANDSLD.113
      IF (HT(IJ) .GT. 0.1) HDIF=AMIN1(HDIF,HRCV)                        LANDSLD.114
      HS(IJ)=HS(IJ)+HDIF                                                LANDSLD.115
      D(IJ)=D(IJ)-HDIF                                                  LANDSLD.116
      E(IJ)=E(IJ)+HDIF                                                  LANDSLD.117
      HSUM=HSUM+SIGN(AMIN1(ABS(HDIF),HT(IJ)),HDIF)                      LANDSLD.118
C                                                                       LANDSLD.119
C     ADJUST WATER VELOCITIES DUE TO DRAGS                              LANDSLD.120
C                                                                       LANDSLD.121
      IJM=IJ-1                                                          LANDSLD.122
      IMJ=MAP(I-1,J)                                                    LANDSLD.123
      IPJ=MAP(I+1,J)                                                    LANDSLD.124
      IF ((IMJ .LE. 0) .OR. (IPJ .LE. 0)) GOTO 400                      LANDSLD.125
      IF ((IMJ .GE. MAXCP) .OR. (IPJ .GE. MAXCP)) GOTO 400              LANDSLD.126
      VIJ=VSA(IJ)                                                       LANDSLD.127
      VSLX=VIJ*COST                                                     LANDSLD.128
      VSLXA=ABS(VSLX)                                                   LANDSLD.129
      VSLY=VIJ*SINT                                                     LANDSLD.130
      VSLYA=ABS(VSLY)                                                   LANDSLD.131
      TLM=CDV/AMAX1(AMIN1(HT(IJ),HIJ),0.01)                             LANDSLD.132
      TEMU=AMIN1(VSLXA*(CDP/DX+TEM),1.)*VSLX                            LANDSLD.133
      TEMV=AMIN1(VSLYA*(CDP/DY+TEM),1.)*VSLY                            LANDSLD.134
      UIJ=U(IJ)                                                         LANDSLD.135
      IF(UIJ.EQ.0.) GO TO 300                                           LANDSLD.136
      VM=(V(IJM)+V(IJ)+V(IPJ-1)+V(IPJ))*0.25                            LANDSLD.137
      UN(IJ)=(AMIN1(SQRT(UIJ*UIJ+VM*VM)/(C(IJ)+C(IPJ)),1.)*UIJ-TEMU)*0.5LANDSLD.138
     **UN(IJ)                                                           LANDSLD.139
  300 UIJ=U(IMJ)                                                        LANDSLD.140
```

```
      IF(UIJ.EQ.0.) GO TO 310                                           LANDSLD.141
      VM=(V(IMJ-1)+V(IMJ)+V(IJM)+V(IJ))*0.25                            LANDSLD.142
      UN(IMJ)=(AMIN1(SQRT(UIJ*UIJ+VM*VM)/(C(IMJ)+C(IJ)),1.)*UIJ-TEMU)*.5LANDSLD.143
     *+UN(IMJ)                                                          LANDSLD.144
  310 VIJ=V(IJ)                                                         LANDSLD.145
      IF(VIJ.EQ.0.) GO TO 320                                           LANDSLD.146
      UM=(U(IMJ)+U(IMJ+1)+U(IJ)+U(IJ+1))*0.25                           LANDSLD.147
      VN(IJ)=(AMIN1(SQRT(UM*UM+VIJ*VIJ)/(C(IJ)+C(IJ+1)),1.)*VIJ         LANDSLD.148
     *-TEMV)*0.5+VN(IJ)                                                 LANDSLD.149
  320 VIJ=V(IJM)                                                        LANDSLD.150
      IF(VIJ.EQ.0.) GO TO 330                                           LANDSLD.151
      UM=(U(IMJ-1)+U(IMJ)+U(IJM)+U(IJ))*0.25                            LANDSLD.152
      VN(IJM)=(AMIN1(SQRT(UM*UM* VIJ*VIJ)/(C(IJM)+C(IJ)),1.)*VIJ-TEMV)  LANDSLD.153
     **0.5+VN(IJM)                                                      LANDSLD.154
  330 CONTINUE                                                          LANDSLD.155
  400 CONTINUE                                                          LANDSLD.156
      SS(IES)=HSUM*DX*DY                                                LANDSLD.157
      RETURN                                                            LANDSLD.158
      END                                                               LANDSLD.159
```

```
      SUBROUTINE SHOTPRT(S,D,MAXCP,MAP,M,N,T,NLBL,LBLP,IPAGP)
C
C     PRINT SNAPSHOT FOR WATER SURFACE ELEVATION OR VELOCITY
C
C     EXTERNALS REFERENCED
C          (NONE)
C     EXTERNALS CONTAINING REFERENCES
C          LSWAVE  (MAIN PROGRAM)
C          SIMULAT
C
      DIMENSION S(MAXCP),D(MAXCP),MAP(M,N),LBLP(1)
      DIMENSION LINE(120),NUM(10),IDIG(6),LBL(4)
      DIMENSION ISYBA(3)
      DATA NUM/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/
      DATA ISYBA/3H...,3H///,3HDDD/
      DATA LWID,LINC/30,4/
      DATA NXLN/5/
      DATA NPL/0/
C
      DO 10 I=1,4
   10 LBL(I)=1H
      DO 20 I=1,NLBL
   20 LBL(I)=LBLP(I)
      LN=LWID*LINC
      NSET=(N-1)/LWID+1
      IPAGE=IPAGP
      IF ((NXLN +M)*NSET+NPL+4 .GT. 63) IPAGE=0
      IF (IPAGE .EQ. 0) NPL=64
      N2=0
      NXLST=0
      DO 160 ISET=1,NSET
      N1=N2+1
      N2=MIN0(N,N2+LWID)
      IF (ISET .EQ. NSET) NXLST=4
      NPL=NPL+NXLN+M+NXLST
      IFF=1H
      IF (NPL .LE. 63) GOTO 30
      NPL=NXLN+M+NXLST
```

```
      IFF=1H1                                                           SHOTPRT.40
   30 IF (ISET .EQ. 1) PRINT 902,IFF,T,LBL,(J,J=N1,N2)                  SHOTPRT.41
      IF (ISET .NE. 1) PRINT 903,IFF,(J,J=N1,N2)                        SHOTPRT.42
      PRINT 904                                                         SHOTPRT.43
      DO 150 I=1,M                                                      SHOTPRT.44
      DO 40 K=1,LN                                                      SHOTPRT.45
   40 LINE(K)=1H                                                        SHOTPRT.46
      L=-LINC                                                           SHOTPRT.47
      DO 140 J=N1,N2                                                    SHOTPRT.48
      L=L+LINC                                                          SHOTPRT.49
      ISYB=ISYBA(3)                                                     SHOTPRT.50
      IJ=MAP(I,J)                                                       SHOTPRT.51
      IF (IJ .LE. 0) GOTO 120                                           SHOTPRT.52
      IF (IJ .GE. MAXCP) GOTO 120                                       SHOTPRT.53
      SIJ=S(IJ)                                                         SHOTPRT.54
      IF ((SIJ .EQ. 0.) .AND. (D(IJ) .EQ. 0.)) GOTO 120                 SHOTPRT.55
      ISYB=ISYBA(1)                                                     SHOTPRT.56
      IF (SIJ) 50,120,60                                                SHOTPRT.57
   50 SIJ=-SIJ                                                          SHOTPRT.58
      LINE(L+1)=1H-                                                     SHOTPRT.59
   60 ISYB=ISYBA(2)                                                     SHOTPRT.60
      KSIJ=SIJ*1000.                                                    SHOTPRT.61
      IF (KSIJ .GE. 999500) GOTO 120                                    SHOTPRT.62
      KRND=5                                                            SHOTPRT.63
      IF (KSIJ .GT. 1000) KRND=50                                       SHOTPRT.64
      IF (KSIJ .GT. 10000) KRND=500                                     SHOTPRT.65
      KSIJ=KSIJ+KRND                                                    SHOTPRT.66
      DO 70 K=1,5                                                       SHOTPRT.67
      KSIJ=KSIJ/10                                                      SHOTPRT.68
   70 IDIG(K)=NUM(KSIJ-(KSIJ/10)*10+1)                                  SHOTPRT.69
      DO 80 K=1,3                                                       SHOTPRT.70
   80 IDIG(7-K)=IDIG(6-K)                                               SHOTPRT.71
      IDIG(3)=1H.                                                       SHOTPRT.72
      DO 90 K=1,3                                                       SHOTPRT.73
   90 IF (IDIG(7-K) .NE. 1H0) GOTO 100                                  SHOTPRT.74
      K=4                                                               SHOTPRT.75
  100 ISHFT=9-K                                                         SHOTPRT.76
      DO 110 K=2,4                                                      SHOTPRT.77
```

```
  110 LINE(L+K)=IDIG(ISHFT-K)                                          SHOTPRT.78
      GOTO 140                                                         SHOTPRT.79
  120 DO 130 K=2,4                                                     SHOTPRT.80
  130 LINE(L+K)=ISYB                                                   SHOTPRT.81
  140 CONTINUE                                                         SHOTPRT.82
      L=L+LINC                                                         SHOTPRT.83
      PRINT 905,I,(LINE(K),K=1,L)                                      SHOTPRT.84
  150 CONTINUE                                                         SHOTPRT.85
  160 CONTINUE                                                         SHOTPRT.86
      NPL=NPL+2                                                        SHOTPRT.87
      PRINT 907,ISYBA                                                  SHOTPRT.88
      IF (NPL .LE. 63) PRINT 908                                       SHOTPRT.89
      RETURN                                                           SHOTPRT.90
C                                                                      SHOTPRT.91
  902 FORMAT (A1/8H TIME = ,F9.3,6H (SEC),10X4A10//6H      J,30I4)     SHOTPRT.92
  903 FORMAT (A1///6H      J,30I4)                                     SHOTPRT.93
  904 FORMAT (3H  I)                                                   SHOTPRT.94
  905 FORMAT (2XI2,2X120A1)                                            SHOTPRT.95
  907 FORMAT (/7XA3,16H  = 0. (EXACTLY)                                SHOTPRT.96
     A      /7XA3,31H  OUT OF RANGE (ABS .GE. 999.5)                   SHOTPRT.97
     B      /7XA3,20H  DRY (NOT COMPUTED))                             SHOTPRT.98
  908 FORMAT (/)                                                       SHOTPRT.99
      END                                                              SHOTPRT.100
```

```
      SUBROUTINE HISTPRT(ES,US,VS,Q,SS,NESP,MESP)                       HISTPRT.2
C                                                                       HISTPRT.3
C     PRINT TIME HISTORIES (SURFACE ELEVATION, VELOCITY, OVER-TOPPING,  HISTPRT.4
C       DISPLACED VOLUME)                                               HISTPRT.5
C                                                                       HISTPRT.6
C     EXTERNALS REFERENCED                                              HISTPRT.7
C          HST1                                                         HISTPRT.8
C          HST2                                                         HISTPRT.9
C          MASK     (CDC INTRINSIC)                                     HISTPRT.10
C          SHIFT    (CDC INTRINSIC)                                     HISTPRT.11
C     EXTERNALS CONTAINING REFERENCES                                   HISTPRT.12
C          LSWAVE   (MAIN PROGRAM)                                      HISTPRT.13
C                                                                       HISTPRT.14
      COMMON C(1350),E(1350),EH(1300),H(1350),HT(1350),H2U(1350),       //.2
     A       H2V(1350),U(1350),UN(1350),V(1350),VN(1350)                //.3
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A       HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,             /DATA/.3
     B         IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,             /DATA/.4
     C       I1OB(4),I2OB(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),       /DATA/.5
     D       JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),       /DATA/.6
     E       J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR    /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A       DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,   /SIMU/.3
     B       GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50), /SIMU/.4
     C       KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                /SIMU/.5
      DIMENSION ES(NESP,MESP),US(NESP,MESP),VS(NESP,MESP),              HISTPRT.18
     A       Q(NESP),SS(NESP)                                           HISTPRT.19
      DIMENSION Z(10),NH(5)                                             HISTPRT.20
      DIMENSION ILB(153),IPT(150),NUM(10)                               HISTPRT.21
      DATA NUM/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/                 HISTPRT.22
      DATA NL/10/                                                       HISTPRT.23
C                                                                       HISTPRT.24
      NMES=0                                                            HISTPRT.25
      DO 5 I=1,5                                                        HISTPRT.26
    5 NH(I)=0                                                           HISTPRT.27
      IF (ILR(2,1) .EQ. 0) GOTO 20                                      HISTPRT.28
      NH(1)=NMES+1                                                      HISTPRT.29
      DO 10 J=1,MES                                                     HISTPRT.30
```

```
      IPT(J)=J                                                          HISTPRT.31
   10 ILB(J)=5H  -E-                                                    HISTPRT.32
      NMES=NMES+MES                                                     HISTPRT.33
   20 IF (ILR(2,2) .EQ. 0) GOTO 30                                      HISTPRT.34
      NH(2)=NMES+1                                                      HISTPRT.35
      NH(3)=NMES+MFS+1                                                  HISTPRT.36
      DO 25 J=1,MES                                                     HISTPRT.37
      NMES=NMES+1                                                       HISTPRT.38
      NMESX=NMES+MFS                                                    HISTPRT.39
      IPT(NMES)=J                                                       HISTPRT.40
      IPT(NMESX)=J                                                      HISTPRT.41
      ILB(NMESX)=5H  -V-                                                HISTPRT.42
      ILB(NMES)=5H  -U-                                                 HISTPRT.43
   25 CONTINUE                                                          HISTPRT.44
      NMES=NMESX                                                        HISTPRT.45
   30 IF (ILR(2,3) .EQ. 0) GOTO 40                                      HISTPRT.46
      NH(4)=NMES+1                                                      HISTPRT.47
      NMES=NMES+1                                                       HISTPRT.48
      CALL HST2(Q,IES,REL,IQ,10.)                                       HISTPRT.49
      ILB(NMES)=7HQ *10** .AND. SHIFT(.NOT.MASK(6),18)                  HISTPRT.50
     A      .OR. SHIFT(NUM(IQ+1).AND.MASK(6),18)                       HISTPRT.51
      RELINV=1./REL                                                     HISTPRT.52
   40 IF (NMES .EQ. 0) RETURN                                           HISTPRT.53
      IF (LSKEY .EQ. 1) GOTO 50                                         HISTPRT.54
      IF (LSKEY .GT. 3) GOTO 50                                         HISTPRT.55
      NH(5)=NMES+1                                                      HISTPRT.56
      NMES=NMES+1                                                       HISTPRT.57
      CALL HST2(SS,IES,SREL,ISS,1000.)                                  HISTPRT.58
      SREL=1./SREL                                                      HISTPRT.59
      ILB(NMES)=9HDISPL VOL                                             HISTPRT.60
      ILB(NMES+1)=6H *10** .AND. SHIFT(.NOT.MASK(6),24)                 HISTPRT.61
     A      .OR. SHIFT(NUM(ISS+1).AND.MASK(6),24)                      HISTPRT.62
   50 CONTINUE                                                          HISTPRT.63
      NP=(NMES-1)/NL+1                                                  HISTPRT.64
      N2=0                                                              HISTPRT.65
      DO 80 IP=1,NP                                                     HISTPRT.66
      N1=N2+1                                                           HISTPRT.67
      N2=MIN0(N2+NL,NMES)                                               HISTPRT.68
```

```
      N21=N2                                                            HISTPRT.69
      DO 60 I=4,5                                                       HISTPRT.70
   60 IF ((N2 .GE. NH(I)) .AND. (N1 .LE. NH(I))) N21=N21-1              HISTPRT.71
      IF (N21 .LT. N1) PRINT 901                                        HISTPRT.72
      IF (N21 .LT. N1) PRINT 905                                        HISTPRT.73
      IF (N21 .GE. N1) PRINT 901,(IPT(J),J=N1,N21)                      HISTPRT.74
      IF (N21 .GE. N1) PRINT 905,(1H(,IHIST(IPT(J)),JHIST(IPT(J)),      HISTPRT.75
     A     1H),J=N1,N21)                                                HISTPRT.76
      N21=N2                                                            HISTPRT.77
      IF (N2 .EQ. NH(5)) N21=N21+1                                      HISTPRT.78
      PRINT 902,(ILB(J),J=N1,N21)                                       HISTPRT.79
      PRINT 904                                                         HISTPRT.80
      N21=N2-N1+1                                                       HISTPRT.81
      DTI=DT*IS*ISP                                                     HISTPRT.82
      TI=TZERO-DTI                                                      HISTPRT.83
      DO 70 I=1,IES,ISP                                                 HISTPRT.84
      CALL HST1(ES,NES,MES,Z,I,1.,NH(1),MES,N1,NL)                      HISTPRT.85
      CALL HST1(US,NES,MES,Z,I,1.,NH(2),MES,N1,NL)                      HISTPRT.86
      CALL HST1(VS,NES,MES,Z,I,1.,NH(3),MES,N1,NL)                      HISTPRT.87
      CALL HST1(Q,NES,1,Z,I,RELINV,NH(4),1,N1,NL)                       HISTPRT.88
      CALL HST1(SS,NES,1,Z,I,SREL,NH(5),1,N1,NL)                        HISTPRT.89
      TI=TI+DTI                                                         HISTPRT.90
      PRINT 903,(I-1)*IS,TI,(Z(J),J=1,N21)                              HISTPRT.91
   70 CONTINUE                                                          HISTPRT.92
   80 CONTINUE                                                          HISTPRT.93
      RETURN                                                            HISTPRT.94
C                                                                       HISTPRT.95
  901 FORMAT (1H1//23X10I9)                                             HISTPRT.96
  902 FORMAT (6X4HSTEP,6X5H(SEC),7X11A9)                                HISTPRT.97
  903 FORMAT (6XI4,1XF10.3,5X10(1XF8.3))                                HISTPRT.98
  904 FORMAT (1H )                                                      HISTPRT.99
  905 FORMAT (6X4HTIME,6X4HTIME,6X10(2XA1,I2,1H,I2,A1))                 HISTPRT.100
      END                                                               HISTPRT.101
```

```
      SUBROUTINE PLOTPRT(PTAB,NESP,MESP,PMAXP,IFLG,ISIZ,NLBL,LBLP)      PLOTPRT.2
C                                                                       PLOTPRT.3
C     LINE PRINTER PLOTS OF TIME HISTORIES                              PLOTPRT.4
C                                                                       PLOTPRT.5
C     EXTERNALS REFERENCED                                              PLOTPRT.6
C          HST2                                                         PLOTPRT.7
C     EXTERNALS CONTAINING REFERENCES                                   PLOTPRT.8
C          LSWAVE  (MAIN PROGRAM)                                       PLOTPRT.9
C                                                                       PLOTPRT.10
      COMMON ZTAB(1000),NBD(1000),MBD(1000),NT2(150)                    PLOTPRT.11
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A     HWEIR1(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,               /DATA/.3
     B       IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,               /DATA/.4
     C     I1OB(4),I2OB(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),         /DATA/.5
     D     JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),         /DATA/.6
     E     J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR      /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A     DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,     /SIMU/.3
     B     GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C     KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                  /SIMU/.5
      DIMENSION PTAB(NESP,MESP)                                         PLOTPRT.14
      DIMENSION LBLP(1),LBL(4)                                          PLOTPRT.15
      DIMENSION NUMS(10)                                                PLOTPRT.16
      DATA NUMS/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/                PLOTPRT.17
      DATA REL,NDIV,NH,NVD,NLAB/1.,1,101,5,2/                           PLOTPRT.18
      DATA NPL/0/                                                       PLOTPRT.19
C                                                                       PLOTPRT.20
      IF (ISIZ .EQ. 0) RETURN                                           PLOTPRT.21
      NCNT=IES                                                          PLOTPRT.22
      NLAB=2                                                            PLOTPRT.23
      IF (ISIZ .EQ. 2) NLAB=5                                           PLOTPRT.24
      NPL=0                                                             PLOTPRT.25
      DO 10 I=1,4                                                       PLOTPRT.26
   10 LBL(I)=1H                                                         PLOTPRT.27
      IF (NLBL .EQ. 0) GOTO 12                                          PLOTPRT.28
      DO 11 I=1,NLBL                                                    PLOTPRT.29
   11 LBL(I)=LBLP(I)                                                    PLOTPRT.30
   12 CONTINUE                                                          PLOTPRT.31
```

```
      PMAX=PMAXP                                                        PLOTPRT.32
      IF ((PMAX .GT. 0.) .OR. (IFLG .EQ. 1)) GOTO 40                    PLOTPRT.33
      DO 30 NP=1,MESP                                                   PLOTPRT.34
      DO 30 I=1,NCNT                                                    PLOTPRT.35
   30 PMAX=AMAX1(PMAX,PTAB(I,NP))                                       PLOTPRT.36
   40 CONTINUE                                                          PLOTPRT.37
      IH=NH+4                                                           PLOTPRT.38
      XMAX=T                                                            PLOTPRT.39
      IF (XMAX .LE. 10.) GOTO 145                                       PLOTPRT.40
      DO 140 I=1,1000                                                   PLOTPRT.41
      XDIV=XMAX/NDIV                                                    PLOTPRT.42
      IDEL=MAX0(1,NH/INT(XDIV))                                         PLOTPRT.43
      MAXX=INT(XDIV+0.5)                                                PLOTPRT.44
      IF (IDEL/10*10 .EQ. IDEL) GOTO 145                                PLOTPRT.45
  140 NDIV=NDIV+1                                                       PLOTPRT.46
  145 CONTINUE                                                          PLOTPRT.47
      DO 490 NP=1,MESP                                                  PLOTPRT.48
      DO 150 I=1,NCNT                                                   PLOTPRT.49
  150 IF (ABS(PTAB(I,NP)) .GT. 0.001) GOTO 160                          PLOTPRT.50
      GOTO 490                                                          PLOTPRT.51
  160 K=I                                                               PLOTPRT.52
C...  SET K=1 TO DISABLE SHIFTING ALONG TIME AXIS                       PLOTPRT.53
      K=1                                                               PLOTPRT.54
      MINX=TZERO/NDIV                                                   PLOTPRT.55
      MAXX=XMAX/NDIV+0.5                                                PLOTPRT.56
      NDIF=MAXX-MINX                                                    PLOTPRT.57
      IDEL=NH/NDIF                                                      PLOTPRT.58
      XMIN=MINX*NDIV                                                    PLOTPRT.59
      NPL=NPL+1                                                         PLOTPRT.60
      NPMOD=MOD(NPL,2)                                                  PLOTPRT.61
      IF (ISIZ .EQ. 2) NPMOD=1                                          PLOTPRT.62
      IF (NPMOD .EQ. 1) PRINT 910                                       PLOTPRT.63
      IF (NPMOD .EQ. 0) PRINT 911                                       PLOTPRT.64
      IF (IFLG .EQ. 0) GOTO 170                                         PLOTPRT.65
      CALL HST2(PTAB,NCNT,REL,IQ,100.)                                  PLOTPRT.66
      RELINV=1./REL                                                     PLOTPRT.67
      PRINT 931,LBL,REL                                                 PLOTPRT.68
      PRINT 932                                                         PLOTPRT.69
```

```
      GOTO 180                                                          PLOTPRT.70
  170 RELINV=1.                                                         PLOTPRT.71
      PRINT 930,LBL,NP,IHIST(NP),JHIST(NP)                              PLOTPRT.72
      PRINT 932                                                         PLOTPRT.73
  180 CONTINUE                                                          PLOTPRT.74
      YMAX=0.                                                           PLOTPRT.75
      DO 190 I=K,NCNT                                                   PLOTPRT.76
      ZTAB(I)=PTAB(I,NP)*RELINV                                         PLOTPRT.77
  190 YMAX=AMAX1(YMAX,ABS(ZTAB(I)))                                     PLOTPRT.78
      IF (PMAX .GT. 0.) YMAX=PMAX                                       PLOTPRT.79
      LABX=YMAX+0.99                                                    PLOTPRT.80
      XLAB=LABX                                                         PLOTPRT.81
      IF (LABX .GE. 1) GOTO 230                                         PLOTPRT.82
      XLAB=0.1                                                          PLOTPRT.83
      IF (YMAX-0.1) 210,230,220                                         PLOTPRT.84
  210 LABX=(YMAX+0.0099)*100.                                           PLOTPRT.85
      LABX=(LABX+1)/2*2                                                 PLOTPRT.86
      XLAB=LABX*0.01                                                    PLOTPRT.87
      GOTO 230                                                          PLOTPRT.88
  220 LABX=(YMAX+0.099)*10.                                             PLOTPRT.89
      XLAB=LABX*0.1                                                     PLOTPRT.90
  230 DELL=XLAB/NLAB                                                    PLOTPRT.91
      DO 240 I=1,IH                                                     PLOTPRT.92
  240 NT2(I)=1H                                                         PLOTPRT.93
      LABX=MINX-1                                                       PLOTPRT.94
      DO 250 L=1,IH,IDEL                                                PLOTPRT.95
      LABX=LABX+1                                                       PLOTPRT.96
      NUMB=LABX*NDIV                                                    PLOTPRT.97
      ISCALE=10000                                                      PLOTPRT.98
      IX=0                                                              PLOTPRT.99
      DO 245 NN=1,4                                                     PLOTPRT.100
      NUMB=NUMB-IX*ISCALE                                               PLOTPRT.101
      ISCALE=0.1*ISCALE                                                 PLOTPRT.102
      IX=NUMB/ISCALE                                                    PLOTPRT.103
  245 NT2(L+NN-1)=NUMS(IX+1)                                            PLOTPRT.104
      IF (NT2(L) .EQ. 1H0) NT2(L)=1H                                    PLOTPRT.105
  250 CONTINUE                                                          PLOTPRT.106
      PRINT 940,(NT2(I),I=1,IH)                                         PLOTPRT.107
```

```
      DYINV=NVD/DELL                                                    PLOTPRT.108
      DXINV=FLOAT(IDEL)/NDIV                                            PLOTPRT.109
      DO 280 I=K,NCNT                                                   PLOTPRT.110
      NBD(I)=(TZERO+(I-1)*DT*IS-XMIN)*DXINV+1.5                         PLOTPRT.111
  280 MBD(I)=(XLAB-ZTAB(I))*DYINV+1.5                                   PLOTPRT.112
      JEND=NVD+1                                                        PLOTPRT.113
      JS=0                                                              PLOTPRT.114
      XLA=XLAB+DELL                                                     PLOTPRT.115
      DO 390 IP=1,2                                                     PLOTPRT.116
      DO 380 I=1,NLAB                                                   PLOTPRT.117
      DO 350 J=1,JEND                                                   PLOTPRT.118
      JS=JS+1                                                           PLOTPRT.119
      JMID=1/JS-J/JEND                                                  PLOTPRT.120
      DO 302 L=1,NH                                                     PLOTPRT.121
  302 NT2(L)=1H                                                         PLOTPRT.122
      IF (JS .EQ. 1) GOTO 304                                           PLOTPRT.123
      IF (I .LT. NLAB) GOTO 310                                         PLOTPRT.124
      IF (J .LT. JEND) GOTO 310                                         PLOTPRT.125
  304 DO 306 L=1,NH                                                     PLOTPRT.126
  306 NT2(L)=1H-                                                        PLOTPRT.127
      DO 308 L=1,NH,IDEL                                                PLOTPRT.128
  308 NT2(L)=1H+                                                        PLOTPRT.129
  310 DO 320 L=K,NCNT                                                   PLOTPRT.130
      IF (MBD(L) .NE. JS) GOTO 320                                      PLOTPRT.131
      NT2(NBD(L))=1H*                                                   PLOTPRT.132
  320 CONTINUE                                                          PLOTPRT.133
      IF (JMID .NE. 0) GOTO 330                                         PLOTPRT.134
      NT2(1)=1HI                                                        PLOTPRT.135
      NT2(NH)=1HI                                                       PLOTPRT.136
      PRINT 970,(NT2(L),L=1,NH)                                         PLOTPRT.137
      GOTO 350                                                          PLOTPRT.138
  330 NT2(1)=1H+                                                        PLOTPRT.139
      IF ((JS .GT. 1) .AND. (I .LT. NLAB)) NT2(NH)=1H+                  PLOTPRT.140
      XLA=XLA-DELL                                                      PLOTPRT.141
      PRINT 960,XLA,(NT2(L),L=1,NH)                                     PLOTPRT.142
  350 CONTINUE                                                          PLOTPRT.143
  380 JEND=NVD                                                          PLOTPRT.144
  390 CONTINUE                                                          PLOTPRT.145
```

```
  490 CONTINUE                                                          PLOTPRT.146
      RETURN                                                            PLOTPRT.147
C                                                                       PLOTPRT.148
  910 FORMAT (1H1)                                                      PLOTPRT.149
  911 FORMAT (2(/))                                                     PLOTPRT.150
  930 FORMAT (2(/),8X4A10,43X9HLOCATION ,I2,3X6HGRID (,I2,1H,I2,1H)/)   PLOTPRT.151
  931 FORMAT (2(/),8X4A10,48X13HMULTIPLIER = ,1PE8.1/)                  PLOTPRT.152
  932 FORMAT (55X14HTIME (SECONDS))                                     PLOTPRT.153
  940 FORMAT (10X,122A1)                                                PLOTPRT.154
  960 FORMAT (1XF10.2,1X120A1)                                          PLOTPRT.155
  970 FORMAT (12X120A1)                                                 PLOTPRT.156
      END                                                               PLOTPRT.157
```

```
      SUBROUTINE ITL1(NWEIR,JWEIR,I1WEIR,I2WEIR,                        ITL1.2
     A     MWEIR,IWEIR,J1WEIR,J2WEIR,DY5)                               ITL1.3
C                                                                       ITL1.4
C     COUNT DAM SEGMENTS, INITIALIZE DY5 (USED IN OVER-TOPPING CALCS)   ITL1.5
C                                                                       ITL1.6
C     EXTERNALS REFERENCED                                              ITL1.7
C          (NONE)                                                       ITL1.8
C     EXTERNALS CONTAINING REFERENCES                                   ITL1.9
C          INITIAL                                                      ITL1.10
C                                                                       ITL1.11
      DIMENSION JWEIR(1),I1WEIR(1),I2WEIR(1)                            ITL1.12
      DIMENSION IWEIR(1),J1WEIR(1),J2WEIR(1)                            ITL1.13
C                                                                       ITL1.14
      IF (NWEIR .LE. 0) RETURN                                          ITL1.15
      NTOTJ=NENDJ=0                                                     ITL1.16
      DO 20 J=1,NWEIR                                                   ITL1.17
      KJ=I2WEIR(J)-I1WEIR(J)+1                                          ITL1.18
      NTOTJ=NTOTJ+KJ                                                    ITL1.19
      KJ1=MIN0(KJ,2)                                                    ITL1.20
      IEND=I1WEIR(J)                                                    ITL1.21
      DO 20 KJ=1,KJ1                                                    ITL1.22
      KEND=1                                                            ITL1.23
      IF (MWEIR .LE. 0) GOTO 15                                         ITL1.24
      DO 10 I=1,MWEIR                                                   ITL1.25
      JEND=J1WEIR(I)                                                    ITL1.26
      DO 10 KI=1,2                                                      ITL1.27
      IF ((IWEIR(I) .EQ. IEND) .AND. (JEND .EQ. JWEIR(J))) KEND=0       ITL1.28
      IF (((IWEIR(I) .EQ. IEND-1) .OR. (IWEIR(I) .EQ. IEND+1))         ITL1.29
     A     .AND. ((JEND .EQ. JWEIR(J)-1) .OR. (JEND .EQ. JWEIR(J)+1)))  ITL1.30
     B          KEND=0                                                 ITL1.31
   10 JEND=J2WEIR(I)                                                    ITL1.32
   15 NENDJ=NENDJ+KEND                                                  ITL1.33
   20 IEND=I2WEIR(J)                                                    ITL1.34
      DY5=DY5*(1.-NENDJ/(2.*NTOTJ))                                     ITL1.35
      RETURN                                                            ITL1.36
      END                                                               ITL1.37
```

```
      SUBROUTINE PRT1(ITYP,NAME,IVAL,ILEN,TEXT)                         PRT1.2
C                                                                       PRT1.3
C     PRINT THE NAME, VALUE, AND DESCRIPTION OF A VARIABLE              PRT1.4
C                                                                       PRT1.5
C     EXTERNALS REFERENCED                                              PRT1.6
C          (NONE)                                                       PRT1.7
C     EXTERNALS CONTAINING REFERENCES                                   PRT1.8
C          DATAPRT                                                      PRT1.9
C          INITNAT                                                      PRT1.10
C          INITFOR                                                      PRT1.11
C                                                                       PRT1.12
      INTEGER TEXT(1)                                                   PRT1.13
      INTEGER FMT(3),FMT1(5),FMT2(10)                                   PRT1.14
      DATA FMT/2*1H ,8H,4X8A10)/                                        PRT1.15
      DATA FMT1/10H(11XA10,  ,10H(1XA10,    ,3*1H /                     PRT1.16
      DATA FMT2/10HG16.6     ,10HI12,4X     ,10HF12.3,4X  ,10H4XA10,2X  ,PRT1.17
     A     6*1H /                                                      PRT1.18
C                                                                       PRT1.19
      FMT(1)=FMT1(ITYP-1)/20+1)                                         PRT1.20
      FMT(2)=FMT2(MOD(ITYP,10))                                         PRT1.21
      LEN=MAX0(1,MIN0(8,ILEN))                                          PRT1.22
      IF (LEN .EQ. ILEN) PRINT FMT,NAME,IVAL,(TEXT(I),I=1,ILEN)         PRT1.23
      IF (LEN .NE. ILEN) PRINT FMT,NAME,IVAL                            PRT1.24
      RETURN                                                            PRT1.25
      END                                                               PRT1.26
```

```
      SUBROUTINE PRT2(Z,N,M,LBL)                                        PRT2.2
C                                                                       PRT2.3
C     PRINT THE INPUT GRID (DEPTH GRID)                                 PRT2.4
C                                                                       PRT2.5
C     EXTERNALS REFERENCED                                              PRT2.6
C          (NONE)                                                       PRT2.7
C     EXTERNALS CONTAINING REFERENCES                                   PRT2.8
C          DATAPRT                                                      PRT2.9
C                                                                       PRT2.10
      DIMENSION Z(N,M)                                                  PRT2.11
      K6=(N-1)/16+1                                                     PRT2.12
      IN1=1                                                             PRT2.13
      KKL=0                                                             PRT2.14
      DO 50 KK=1,K6                                                     PRT2.15
      IN2=MIN0(IN1+15,N)                                                PRT2.16
      KKL=KKL+M+7                                                       PRT2.17
      IF (KK .EQ. 1) GOTO 20                                            PRT2.18
      KFF=1H0                                                           PRT2.19
      IF (KKL .LE. 63) GOTO 10                                          PRT2.20
      KKL=M+7                                                           PRT2.21
      KFF=1H1                                                           PRT2.22
   10 PRINT 102,KFF,LBL                                                 PRT2.23
      GOTO 30                                                           PRT2.24
   20 PRINT 101,LBL                                                     PRT2.25
   30 PRINT 103,(K,K=IN1,IN2)                                           PRT2.26
      PRINT 104                                                         PRT2.27
      DO 40 J=1,M                                                       PRT2.28
   40 PRINT 105,J,(Z(K,J),K=IN1,IN2)                                    PRT2.29
   50 IN1=IN2+1                                                         PRT2.30
      RETURN                                                            PRT2.31
C                                                                       PRT2.32
  101 FORMAT (1H1,A5,5H GRID)                                           PRT2.33
  102 FORMAT (A1/1HJ,A5,17H GRID (CONTINUED))                           PRT2.34
  103 FORMAT (/5X1HJ,I5,16I8)                                           PRT2.35
  104 FORMAT (2X1HI)                                                    PRT2.36
  105 FORMAT (1XI3,16F8.1)                                              PRT2.37
      END                                                               PRT2.38
```

```
      SUBROUTINE PRT3(Z,N,M)                                            PRT3.2
C                                                                       PRT3.3
C     PRINT AN ARRAY OF CHARACTERS                                      PRT3.4
C                                                                       PRT3.5
C     EXTERNALS REFERENCED                                              PRT3.6
C          (NONE)                                                       PRT3.7
C     EXTERNALS CONTAINING REFERENCES                                   PRT3.8
C          DATAPRT                                                      PRT3.9
C                                                                       PRT3.10
      DIMENSION Z(N,M)                                                  PRT3.11
      NZ=N/10                                                           PRT3.12
      IF (NZ .GE. 1) PRINT 101,(J,J=1,NZ)                               PRT3.13
      IF (NZ .EQ. 0) PRINT 101                                          PRT3.14
      PRINT 102,(MOD(J,10),J=1,N)                                       PRT3.15
      PRINT 103                                                         PRT3.16
      DO 10 I=1,M                                                       PRT3.17
      PRINT 104,I,(Z(J,I),J=1,N)                                        PRT3.18
   10 CONTINUE                                                          PRT3.19
C                                                                       PRT3.20
  101 FORMAT (10X1HJ,9I10)                                              PRT3.21
  102 FORMAT (11X99I1)                                                  PRT3.22
  103 FORMAT (6X1HI)                                                    PRT3.23
  104 FORMAT (6XI2,3X99A1)                                              PRT3.24
      RETURN                                                            PRT3.25
      END                                                               PRT3.26
```

```
      SUBROUTINE PRT4(KVERS)                                            PRT4.2
C                                                                       PRT4.3
C     PRINT HEADING OF THE COMPUTER RUN                                 PRT4.4
C                                                                       PRT4.5
C     EXTERNALS REFERENCED                                              PRT4.6
C          TIME     (CDC LIBRARY)                                      PRT4.7
C          DATE     (CDC LIBRARY)                                      PRT4.8
C          MASK     (CDC INTRINSIC)                                    PRT4.9
C          SHIFT    (CDC INTRINSIC)                                    PRT4.10
C     EXTERNALS CONTAINING REFERENCES                                   PRT4.11
C          DATAROR                                                      PRT4.12
C                                                                       PRT4.13
      DIMENSION ID(80),NCHR(10)                                         PRT4.14
      DIMENSION KHEAD(6,5)                                              PRT4.15
      IDCHR(I)=SHIFT(ID((I-1)/10+1),6*MOD(I-1,10))                      PRT4.16
      DATA ISP,IST/1H ,1H-/                                             PRT4.17
      DATA KHEAD/5HL     ,5H SSS ,5HW   W,5H  A  ,5HV   V,5HEEEEE       PRT4.18
     A           ,5HL     ,5HS    ,5HW   W,5H A A ,5HV   V,5HE          PRT4.19
     B           ,5HL     ,5H SSS ,5HW W W,5HA   A,5HV   V,5HEEE        PRT4.20
     C           ,5HL     ,5H    S,5HW W W,5HAAAAA,5H V V ,5HE          PRT4.21
     D           ,5HLLLLL,5HSSSS ,5H W W ,5HA   A,5H  V  ,5HEEEEE/      PRT4.22
C                                                                       PRT4.23
      CALL TIME(KTIME)                                                  PRT4.24
      CALL DATE(KDATE)                                                  PRT4.25
      NXCHR=0                                                           PRT4.26
      IDN=0                                                             PRT4.27
      IDI=-8                                                            PRT4.28
   90 IDN=IDN+1                                                         PRT4.29
      IDI=IDI+8                                                         PRT4.30
      READ 903,(ID(IDI+I),I=1,8)                                        PRT4.31
      IDCONT=IDCHR(80+IDI*10) .AND. MASK(6)                             PRT4.32
      IDCONT=SHIFT((IDCONT .EQ. 1L+),-60) .OR.                          PRT4.33
     A        SHIFT((IDCONT .EQ. 1L/),-60)                             PRT4.34
      IDCONT=IDCONT .AND..NOT. MASK(59)                                 PRT4.35
      NMCHR=80-IDCONT                                                   PRT4.36
      DO 91 I=1,NMCHR                                                   PRT4.37
      ICHR=(IDCHR(NMCHR+1-I+IDI*10) .AND. MASK(6))                      PRT4.38
      IF (ICHR .NE. 1L ) GOTO 92                                        PRT4.39
```

```
   91 CONTINUE                                                          PRT4.40
      I=NMCHR+1                                                         PRT4.41
   92 NCHR(IDN)=NMCHR+1-I                                               PRT4.42
      NXCHR=MAX0(NXCHR,NCHR(IDN))                                       PRT4.43
      IF ((IDCONT .EQ. 1) .AND. (IDN .LT. 10)) GOTO 90                  PRT4.44
      PRINT 901,(KHEAD(I,1),I=1,6),KVERS,((KHEAD(I,J),I=1,6),J=2,3),    PRT4.45
     A      KTIME,KDATE,((KHEAD(I,J),I=1,6),J=4,5)                     PRT4.46
      IF ((NCHR(1) .EQ. 0) .AND. (IDN .EQ. 1)) GOTO 94                  PRT4.47
      IF (IDN .LE. 5) PRINT 905                                         PRT4.48
      IF (IDN .GT. 5) PRINT 904                                         PRT4.49
      NSPA=(131-12-NXCHR)/2+1                                           PRT4.50
      NXSPB=NSTB=3                                                      PRT4.51
      NSTA=NXCHR+2*NXSPB+2*NSTB                                         PRT4.52
      PRINT 902,(ISP,I=1,NSPA),(IST,I=1,NSTA)                           PRT4.53
      DO 93 J=1,IDN                                                     PRT4.54
      NMCHR=MAX0(1,NCHR(J))                                             PRT4.55
      NSPB=NXSPB+(NXCHR-NMCHR)/2                                        PRT4.56
      NSPC=2*NXSPB+NXCHR-NSPB-NMCHR                                     PRT4.57
   93 PRINT 902,(ISP,I=1,NSPA),(IST,I=1,NSTB),(ISP,I=1,NSPB),           PRT4.58
     A      (IDCHR(I+(J-1)*80),I=1,NMCHR  ),(ISP,I=1,NSPC),(IST,I=1,NSTB)PRT4.59
      PRINT 902,(ISP,I=1,NSPA),(IST,I=1,NSTA)                           PRT4.60
   94 CONTINUE                                                          PRT4.61
      IF (IDN .LE. 5) PRINT 905                                         PRT4.62
      IF (IDN .GT. 5) PRINT 904                                         PRT4.63
      RETURN                                                            PRT4.64
C                                                                       PRT4.65
  901 FORMAT (1H1/1X22H      -- LSWAVE --      26X6A6,                  PRT4.66
     A           25X22H  -- DEVELOPED BY --                             PRT4.67
     B      /1X14H     VERSION  A4,30X6A6,                              PRT4.68
     C           25X22H   TETRA TECH, INC.                              PRT4.69
     D      /24X22(1H-)3X6A6,1X21(1H-)                                  PRT4.70
     E            3X22H630  N. ROSEMEAD BLVD.                           PRT4.71
     F      /1XA10,1XA10,27X6A6,                                        PRT4.72
     G           25X22HPASADENA, CALIF. 91107                           PRT4.73
     H      /49X6A6,                                                    PRT4.74
     I           25X22H    (213) 449-6400    )                         PRT4.75
  902 FORMAT (132A1)                                                    PRT4.76
  903 FORMAT (8A10)                                                     PRT4.77
```

```
904 FORMAT (2(/))                                          PRT4.78
905 FORMAT (4(/))                                          PRT4.79
    END                                                    PRT4.80
```

```
      SUBROUTINE PRT5(Z,N,M)                                            PRT5.2
C                                                                       PRT5.3
C     PRINT ARRAY OF CHARACTERS (ALTERNATE VERSION OF PRT3)             PRT5.4
C                                                                       PRT5.5
C     EXTERNALS REFERENCED                                              PRT5.6
C          (NONE)                                                       PRT5.7
C     EXTERNALS CONTAINING REFERENCES                                   PRT5.8
C          INITFOR                                                      PRT5.9
C          INITNAT                                                      PRT5.10
C          INITFLT                                                      PRT5.11
C                                                                       PRT5.12
      DIMENSION Z(N,M)                                                  PRT5.13
      DIMENSION NUM(10)                                                 PRT5.14
      DATA NUM/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/                 PRT5.15
C                                                                       PRT5.16
      NZ=N/10                                                           PRT5.17
      IF (NZ .GE. 1) PRINT 101,1HJ,(J,J=1,NZ)                           PRT5.18
      IF (NZ .EQ. 0) PRINT 101,1HJ                                      PRT5.19
      PRINT 102,(MOD(J,10),J=1,N)                                       PRT5.20
      PRINT 103,1HI,(1H-,J=1,N)                                         PRT5.21
      DO 10 I=1,M                                                       PRT5.22
      IDIG1=NUM(I/10+1)                                                 PRT5.23
      IDIG2=NUM(MOD(I,10)+1)                                            PRT5.24
      IF (I .GE. 10) GOTO 5                                             PRT5.25
      IDIG1=NUM(I+1)                                                    PRT5.26
      IDIG2=1H                                                          PRT5.27
    5 PRINT 104,I,(Z(J,I),J=1,N),1H-,IDIG1,IDIG2                        PRT5.28
   10 CONTINUE                                                          PRT5.29
      PRINT 103,1H ,(1H-,J=1,N)                                         PRT5.30
      PRINT 102,(MOD(J,10),J=1,N)                                       PRT5.31
      IF (NZ .EQ. 0) PRINT 101,1H                                       PRT5.32
      IF (NZ .GE. 1) PRINT 101,1H ,(J,J=1,NZ)                           PRT5.33
      RETURN                                                            PRT5.34
C                                                                       PRT5.35
  101 FORMAT (10XA1,9I10)                                               PRT5.36
  102 FORMAT (11X99I1)                                                  PRT5.37
  103 FORMAT (8XA1,2X99A1)                                              PRT5.38
  104 FORMAT (8XI2,1H-102A1)                                            PRT5.39
```

END

PRT5.40

209

```
      SUBROUTINE HST1(ZS,NES,MES,Z,IZ,RZ,NHI,NHS,N1,NP)                 HST1.2
C                                                                       HST1.3
C     PICK OFF VALUES NEEDED FOR ONE LINE OF PRINT FOR ROUTINE HISTPRT  HST1.4
C                                                                       HST1.5
C     EXTERNALS REFERENCED                                              HST1.6
C          (NONE)                                                       HST1.7
C     EXTERNALS CONTAINING REFERENCES                                   HST1.8
C          HISTPRT                                                      HST1.9
C                                                                       HST1.10
      DIMENSION ZS(NES,MES),Z(NP)                                       HST1.11
      IF (NHI .EQ. 0) RETURN                                            HST1.12
      IF (NHI+NHS-1 .LT. N1) RETURN                                     HST1.13
      N2=N1+NP-1                                                        HST1.14
      IF (NHI .GT. N2) RETURN                                           HST1.15
      NH1=MAX0(NHI,N1)                                                  HST1.16
      NH2=MIN0(NHI+NHS-1,N2)                                            HST1.17
      DO 10 J=NH1,NH2                                                   HST1.18
   10 Z(J-N1+1)=ZS(IZ,J-NHI+1)+RZ                                       HST1.19
      RETURN                                                            HST1.20
      END                                                               HST1.21
```

```
      SUBROUTINE HST2(Q,NES,REL,IQ,QLIM)                                HST2.2
C                                                                       HST2.3
C     DETERMINE SCALING FACTOR FOR PRINTING OR PLOTTING                 HST2.4
C                                                                       HST2.5
C     EXTERNALS REFERENCED                                              HST2.6
C          (NONE)                                                       HST2.7
C     EXTERNALS CONTAINING REFERENCES                                   HST2.8
C          HISTPRT                                                      HST2.9
C          PLOTPRT                                                      HST2.10
C                                                                       HST2.11
      DIMENSION Q(NES)                                                  HST2.12
      QMAX=0.                                                           HST2.13
      DO 10 I=1,NES                                                     HST2.14
   10 QMAX=AMAX1(QMAX,Q(I))                                             HST2.15
      REL=1.                                                            HST2.16
      IQ=0                                                              HST2.17
      IF (QMAX .LT. QLIM) RETURN                                        HST2.18
      IREL=1                                                            HST2.19
      DO 20 I=1,8                                                       HST2.20
      QMAX=QMAX*0.1                                                     HST2.21
      IREL=IREL*10                                                      HST2.22
      IQ=IQ+1                                                           HST2.23
   20 IF (QMAX .LT. QLIM) GOTO 30                                       HST2.24
      IREL=IREL*10                                                      HST2.25
      IQ=IQ+1                                                           HST2.26
   30 REL=IREL                                                          HST2.27
      RETURN                                                            HST2.28
      END                                                               HST2.29
```

```
      SUBROUTINE SLD1(MAP,MP,NP,XSP,LSP,MSLP,NSLP)                      SLD1.2
C                                                                       SLD1.3
C     FIND COMPUTATIONAL ELEMENTS LYING IN SLIDE PATH                   SLD1.4
C                                                                       SLD1.5
C     EXTERNALS REFERENCED                                              SLD1.6
C          CSQRT                                                        SLD1.7
C          ERR                                                          SLD1.8
C     EXTERNALS CONTAINING REFERENCES                                   SLD1.9
C          INITFOR                                                      SLD1.10
C          INITNAT                                                      SLD1.11
C                                                                       SLD1.12
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A     HWEIR1(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,               /DATA/.3
     B       IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,               /DATA/.4
     C     I10B(4),I20B(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),         /DATA/.5
     D     JHIST(50),JOB(4),J10B(4),J20B(4),JWEIR(9),J1WEIR(9),         /DATA/.6
     E     J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR      /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A     DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,     /SIMU/.3
     B     GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C     KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                  /SIMU/.5
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,          /DSLD/.2
     A     WSL,DSL,XO,YO,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),             /DSLD/.3
     B     XSL(101),TSL(101),HSL(101,10),VSL(101,10)                    /DSLD/.4
      COMMON /SLD1/ NSP(10),NKSP,IKSP(500),JKSP(500),XSN(10),VSN(10),   /SLD1/.2
     A     MA,NA,DXA,DYA,DXDXA,DYDXA,DXDYA,DYDYA,AMN,MNA,ITV,LSMAX(10), /SLD1/.3
     B     XA(2500)                                                     /SLD1/.4
      DIMENSION MAP(MP,NP),XSP(MSLP,NSLP),LSP(MSLP,NSLP)                SLD1.17
      LOGICAL OUTSID                                                    SLD1.18
      INTX(X)=INT(X)-(1.AND.SHIFT(X+1.E-10,-60))                        SLD1.19
      DATA EPS/1.E-12/                                                  SLD1.20
C                                                                       SLD1.21
      DXMDXA=DX-DXA                                                     SLD1.22
      DYMDYA=DY-DYA                                                     SLD1.23
      DYA0=0.5*DYA-DYMDYA                                               SLD1.24
      DXA0=0.5*DXA-DXMDXA                                               SLD1.25
      DO 30 J=1,N                                                       SLD1.26
      DO 30 I=1,M                                                       SLD1.27
```

```
      IJ=MAP(I,J)                                                       SLD1.28
      IF (IJ .LE. 0) GOTO 30                                            SLD1.29
      IF (IJ .GE. MAXCP) GOTO 30                                        SLD1.30
      IF (D(IJ) .EQ. 0.) GOTO 30                                        SLD1.31
      XPMIN=YPMIN=1.E10                                                 SLD1.32
      XPMAX=YPMAX=-1.E10                                                SLD1.33
      X=(I-0.5)*DX+DXA0                                                 SLD1.34
      Y0=(J-0.5)*DY+DYA0                                                SLD1.35
      DO 10 KI=1,2                                                      SLD1.36
      X=X+DXMDXA                                                        SLD1.37
      Y=Y0                                                              SLD1.38
      DO 10 KJ=1,2                                                      SLD1.39
      Y=Y+DYMDYA                                                        SLD1.40
      XP=(X-X0)*COST+(Y-Y0)*SINT                                        SLD1.41
      YP=(X0-X)*SINT+(Y-Y0)*COST                                        SLD1.42
      XPMIN=AMIN1(XPMIN,XP)                                             SLD1.43
      XPMAX=AMAX1(XPMAX,XP)                                             SLD1.44
      YPMIN=AMIN1(YPMIN,YP)                                             SLD1.45
      YPMAX=AMAX1(YPMAX,YP)                                             SLD1.46
   10 CONTINUE                                                          SLD1.47
      K1=MAX0(1,INTX((YPMIN-EPS)/DSL)+1)                                SLD1.48
      K2=MIN0(NSL,INTX((YPMAX+EPS)/DSL)+1)                              SLD1.49
      IF (K2 .LT. K1) GOTO 30                                           SLD1.50
      OUTSID=.TRUE.                                                     SLD1.51
      DO 20 K=K1,K2                                                     SLD1.52
      IF (XPMIN .GT. XSN(K)-EPS) GOTO 20                                SLD1.53
      IF (XPMAX .LT. -(XSL(NXH)+EPS)) GOTO 20                           SLD1.54
      OUTSID=.FALSE.                                                    SLD1.55
      INSP=NSP(K)=NSP(K)+1                                              SLD1.56
      IF (INSP.GT.LSMAX(1)) CALL ERR(52,1,8,80H-- EXCEEDED MAXIMUM NUMBESLD1.57
     ,R OF SLIDE COMPUTATIONAL POINTS IN PATH OF SINGLE SLICE,          SLD1.58
     , 1,2,9HSLICE NO.,K)                                               SLD1.59
      XSP(INSP,K)=XPMIN                                                 SLD1.60
      LSP(INSP,K)=NKSP+1                                                SLD1.61
   20 CONTINUE                                                          SLD1.62
      IF (OUTSID) GOTO 30                                               SLD1.63
      NKSP=NKSP+1                                                       SLD1.64
      IKSP(NKSP)=I                                                      SLD1.65
```

```
      JKSP(NKSP)=J                                                      SLD1.66
   30 CONTINUE                                                          SLD1.67
C                                                                       SLD1.68
      DO 40 K=1,NSL                                                     SLD1.69
   40 CALL CSORT(XSP(1,K),500,1,1,NSP(K),1,2)                           SLD1.70
      RETURN                                                            SLD1.71
      END                                                               SLD1.72
```

```
      SUBROUTINE SLD2                                                   SLD2.2
C                                                                       SLD2.3
C     INITIALIZE GEOMETRIC PARAMETERS FOR NATURAL LANDSLIDE             SLD2.4
C                                                                       SLD2.5
C     EXTERNALS REFERENCED                                              SLD2.6
C          TAN      (CDC LIBRARY)                                      SLD2.7
C     EXTERNALS CONTAINING REFERENCES                                   SLD2.8
C          INITNAT                                                      SLD2.9
C                                                                       SLD2.10
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A     HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,              /DATA/.3
     B       IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,              /DATA/.4
     C     I10B(4),I20B(4),IS,ISR,IWEIR(9),I1WEIR(9),I2WEIR(9),        /DATA/.5
     D     JHIST(50),JOB(4),J10B(4),J20B(4),JWEIR(9),J1WEIR(9),        /DATA/.6
     E     J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR     /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,     /SIMU/.2
     A     DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,    /SIMU/.3
     B     GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C     KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                 /SIMU/.5
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,         /DSLD/.2
     A     WSL,DSL,X0,Y0,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),            /DSLD/.3
     B     XSL(101),YSL(101),HSL(101,10),VSL(101,10)                   /DSLD/.4
      COMMON /SLD4/ NSEG,NSS,KIJ,DS,VTOTAL,HMAX0,XNN(100),HNN(100),    /SLD4/.2
     A     DN(100),VOL(100)                                            /SLD4/.3
      ALINEAR(X,X2,X1,Y2,Y1)=Y1+(X-X1)*(Y2-Y1)/(X2-X1)                  SLD2.15
C                                                                       SLD2.16
C     DEFINE CONSTANTS                                                  SLD2.17
C                                                                       SLD2.18
      SGSL=1./SGSL                                                      SLD2.19
      PHIR=PHI*PI/180.                                                  SLD2.20
      FRIC=TAN(PHIR)                                                    SLD2.21
C                                                                       SLD2.22
C     DEFINE STEP SIZE                                                  SLD2.23
C                                                                       SLD2.24
      TEM=XSL(NXH)-XSL(1)                                               SLD2.25
      NSEG=TEM/AMIN1(DX,DY)+0.5                                         SLD2.26
      NSS=NSEG+1                                                        SLD2.27
      DS=TEM/NSEG                                                       SLD2.28
```

```
      IF(NSEG.GT.100) STOP 100                                          SLD2.29
C                                                                       SLD2.30
C     INTERPOLATE FOR INTERNAL H VS. X TABLE WITH EQUAL INCREMENTS DS   SLD2.31
C                                                                       SLD2.32
      XNN(1)=XSL(1)                                                     SLD2.33
      HNN(1)=HSL(1,1)                                                   SLD2.34
      XNN(NSS)=XSL(NXH)                                                 SLD2.35
      HNN(NSS)=HSL(NXH,1)                                               SLD2.36
      DO 30 I=2,NSEG                                                    SLD2.37
      XNN(I)=(I-1)*DS                                                   SLD2.38
      DO 20 J=2,NXH                                                     SLD2.39
      IF(XNN(I).GE.XSL(J)) GO TO 20                                     SLD2.40
      HNN(I)=ALINEAR(XNN(I),XSL(J),XSL(J-1),HSL(J,1),HSL(J-1,1))        SLD2.41
      GO TO 30                                                          SLD2.42
   20 CONTINUE                                                          SLD2.43
   30 CONTINUE                                                          SLD2.44
C                                                                       SLD2.45
C     CALCULATE CROSS SECTIONAL AREA OF SLIDE                           SLD2.46
C                                                                       SLD2.47
      AREA=0.                                                           SLD2.48
      HMAXO=HSL(1,1)                                                    SLD2.49
      DO 40 I=2,NXH                                                     SLD2.50
      HMAXO=AMAX1(HMAXO,HSL(I,1))                                       SLD2.51
   40 AREA=AREA+(XSL(I)-XSL(I-1))*(HSL(I,1)+HSL(I-1,1))                 SLD2.52
      AREA=0.5*AREA                                                     SLD2.53
C                                                                       SLD2.54
C     CALCULATE CROSS SECTIONAL AREA OF ESTIMATED SLIDE                 SLD2.55
C                                                                       SLD2.56
      AREAI=0.                                                          SLD2.57
      DO 45 I=2,NSS                                                     SLD2.58
   45 AREAI=AREAI+(XNN(I)-XNN(I-1))*(HNN(I)+HNN(I-1))                   SLD2.59
      AREAI=0.5*AREAI                                                   SLD2.60
C                                                                       SLD2.61
C     ADJUST ESTIMATED HEIGHT SO THAT INITIAL VOLUME IS RETAINED        SLD2.62
C                                                                       SLD2.63
      RATIO=AREA/AREAI                                                  SLD2.64
      DO 46 I=1,NSS                                                     SLD2.65
   46 HNN(I)=RATIO*HNN(I)                                               SLD2.66
```

```
C                                                                       SLD2.67
C     DEFINE VOLUME OF EACH SEGMENT OF SLIDE AND VOLUME OF TOTAL SLIDE  SLD2.68
C                                                                       SLD2.69
      VTOTAL=0.                                                         SLD2.70
      TEM=WSL*DS*0.5                                                    SLD2.71
      DO 50 I=1,NSEG                                                    SLD2.72
      VOL(I)=TEM*(HNN(I+1)+HNN(I))                                      SLD2.73
   50 VTOTAL=VTOTAL+VOL(I)                                              SLD2.74
      RETURN                                                            SLD2.75
      END                                                               SLD2.76
```

```
      SUBROUTINE SLD3(MAP,MP,NP)                                        SLD3.2
C                                                                       SLD3.3
C     PICK EQUALLY SPACED POINTS OF INCREMENT DS ALONG THE CENTER       SLD3.4
C       LINE OF NATURAL SLIDE                                           SLD3.5
C                                                                       SLD3.6
C     EXTERNALS REFERENCED                                              SLD3.7
C          ERR                                                          SLD3.8
C     EXTERNALS CONTAINING REFERENCES                                   SLD3.9
C          INITNAT                                                      SLD3.10
C                                                                       SLD3.11
      COMMON /DATA/ D(200),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,          /DATA/.2
     A     HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,               /DATA/.3
     B       IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,               /DATA/.4
     C     I1OB(4),I2OB(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),         /DATA/.5
     D     JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),         /DATA/.6
     E     J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR      /DATA/.7
      COMMON /SIMU/ CC,CDTDX,CDTDX5,CDTDY,CDTDY5,CD2,CTMR,C3X,C3Y,      /SIMU/.2
     A     DTDX,DTDX5,DTDY,DTDY5,DXDT2,DXDT2C,DX5,DYDT2,DYDT2C,DY5,     /SIMU/.3
     B     GDTDX,GDTDY,G5,HWEI1,HWEI2,HWEJ1,HWEJ2,IES,ITS(21),JES(50),  /SIMU/.4
     C     KES,KS,KT,MAXCP,MAXCPM,MAXCPN,MS,NS,PI,QT,T                  /SIMU/.5
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,          /DSLD/.2
     A     WSL,DSL,X0,Y0,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),             /DSLD/.3
     B     XSL(101),TSL(101),HSL(101,10),VSL(101,10)                    /DSLD/.4
      COMMON /SLD4/ NSEG,NSS,KIJ,DS,VTOTAL,HMAXO,XNN(100),HNN(100),     /SLD4/.2
     A     DN(100),VOL(100)                                             /SLD4/.3
      DIMENSION MAP(MP,NP)                                              SLD3.16
      AINTER(P,Y1,Y2) = (Y2-Y1)*P+Y1                                    SLD3.17
C                                                                       SLD3.18
      XMAX=M*DX                                                         SLD3.19
      YMAX=N*DY                                                         SLD3.20
      KIJ=1                                                             SLD3.21
      XP=-NSEG*DS                                                       SLD3.22
      YP=WSL*0.5                                                        SLD3.23
C                                                                       SLD3.24
   10 CONTINUE                                                          SLD3.25
      X=XP*COST-YP*SINT+X0                                              SLD3.26
      Y=XP*SINT+YP*COST+Y0                                              SLD3.27
C                                                                       SLD3.28
```

```
      IF(X.GT.XMAX.OR.X.LT.DX) GO TO 20                                 SLD3.29
      IF(Y.GT.YMAX.OR.Y.LT.DY) GO TO 20                                 SLD3.30
      XI=X/DX                                                           SLD3.31
      I=XI                                                              SLD3.32
      YJ=Y/DY                                                           SLD3.33
      J=YJ                                                              SLD3.34
      IJ=MAP(I,J)                                                       SLD3.35
      IPJ=MAP(I+1,J)                                                    SLD3.36
      IF ((IJ .LE. 0) .OR. (IJ .GE. MAXCP)) GOTO 15                     SLD3.37
      IF ((IPJ .LE. 0) .OR. (IPJ .GE. MAXCP)) GOTO 15                   SLD3.38
      DN(KIJ)=AINTER(YJ-J,AINTER(XI-I,D(IJ),D(IPJ)),                    SLD3.39
     *AINTER(XI-I,D(IJ+1),D(IPJ+1)))                                    SLD3.40
      KIJ=KIJ+1                                                         SLD3.41
      IF (KIJ.GT.100) CALL ERR(54,1,8,71H-- EXCEEDED MAXIMUM NUMBER OF CSLD3.42
     ,OMPUTATIONAL POINTS IN NATURAL LANDSLIDE,0)                       SLD3.43
   15 XP=XP+DS                                                          SLD3.44
      GO TO 10                                                          SLD3.45
   20 KIJ=KIJ-1                                                         SLD3.46
      RETURN                                                            SLD3.47
      END                                                               SLD3.48
```

```
      SUBROUTINE SLD4                                                   SLD4.2
C                                                                       SLD4.3
C     GENERATE TABLE OF SLIDE VELOCITY VS. TIME FOR UNEQUAL TIME STEP   SLD4.4
C       (NATURAL LANDSLIDE)                                             SLD4.5
C                                                                       SLD4.6
C     EXTERNALS REFERENCED                                              SLD4.7
C          SIN      (CDC LIBRARY)                                      SLD4.8
C          COS      (CDC LIBRARY)                                      SLD4.9
C          ATAN     (CDC LIBRARY)                                      SLD4.10
C          SQRT     (CDC LIBRARY)                                      SLD4.11
C     EXTERNALS CONTAINING REFERENCES                                   SLD4.12
C          INITNAT                                                      SLD4.13
C                                                                       SLD4.14
      COMMON /DATA/ D(2000),CD,CMANN,CED,DCONV,DT,DX,DY,EMAX,G,         /DATA/.2
     A     HWEIRI(9),HWEIRJ(9),TMAX,TS(21),TZERO,UMAX,Z0,              /DATA/.3
     B       IC,ICC,IENV,IHIST(50),ILR(2,3),IOB(4),IPLTF,              /DATA/.4
     C     I1OB(4),I2OB(4),IS,ISP,IWEIR(9),I1WEIR(9),I2WEIR(9),        /DATA/.5
     D     JHIST(50),JOB(4),J1OB(4),J2OB(4),JWEIR(9),J1WEIR(9),        /DATA/.6
     E     J2WEIR(9),LSE,LSKEY,LSU,M,MES,MOB,MWEIR,N,NES,NOB,NWEIR     /DATA/.7
      COMMON /DSLD/ NSL,MSL,NXH,NTV,MSUB,NSUB,THETA,COST,SINT,          /DSLD/.2
     A     WSL,DSL,XO,YO,CDP,CDV,SGSL,AK,PHI,FRIC,TITLE(7),            /DSLD/.3
     B     XSL(101),TSL(101),HSL(101,10),VSL(101,10)                   /DSLD/.4
      COMMON /SLD4/ NSEG,NSS,KIJ,DS,VTOTAL,HMAXO,XNN(100),HNN(100),     /SLD4/.2
     A     DN(100),VOL(100)                                            /SLD4/.3
C                                                                       SLD4.18
C     LOOP ON POSSIBLE NUMBER OF EQUAL DISTANCE STEPS THE SLIDE MAY MOVESLD4.19
C                                                                       SLD4.20
      TSL(1)=0.                                                         SLD4.21
      VSL(1,1)=0.                                                       SLD4.22
      A=0.                                                              SLD4.23
      VEL=0.                                                            SLD4.24
      KSTEP=KIJ-NSS+1                                                   SLD4.25
      DO 10 IKS=1,KSTEP                                                 SLD4.26
      JS=IKS-1                                                          SLD4.27
      AOLD=A                                                            SLD4.28
      A=0.                                                              SLD4.29
      HMAX=0.                                                           SLD4.30
      VWTOT=0.                                                          SLD4.31
```

```
C                                                                       SLD4.32
C     LOOP ON NUMBER OF SEGMENTS IN SLIDE                               SLD4.33
C                                                                       SLD4.34
      DO 20 I=1,NSEG                                                    SLD4.35
      RHO=SGSL                                                          SLD4.36
C                                                                       SLD4.37
C     GET INDICIES OF A SINGLE SLIDE SEGMENT AND THE CORRESPONDING      SLD4.38
C      INDICIES OF DEPTH                                                SLD4.39
C                                                                       SLD4.40
      I1=I+1                                                            SLD4.41
      J=NSS-I+1+JS                                                      SLD4.42
      J1=J-1                                                            SLD4.43
      DJ=AMAX1(AMIN1(DN(J),HNN(I)),0.)                                  SLD4.44
      DJ1=AMAX1(AMIN1(DN(J1),HNN(I1)),0.)                               SLD4.45
C                                                                       SLD4.46
C     CALCULATE BOTTOM SLOPE                                            SLD4.47
C                                                                       SLD4.48
      ANGLE=ATAN((DN(J)-DN(J1))/DS)                                     SLD4.49
C                                                                       SLD4.50
C     FIND SUBMERGED VOLUME OF EACH SEGMENT                             SLD4.51
C                                                                       SLD4.52
      VW=0.5*WSL*DS*(DJ+DJ1)                                            SLD4.53
      VWTOT=VWTOT+VW                                                    SLD4.54
      IF(VW.EQ.0.) RHO=0.                                               SLD4.55
      HMAX=AMAX1(HMAX,DJ,DJ1)                                           SLD4.56
      HMAX=AMIN1(HMAXG,HMAX)                                            SLD4.57
C                                                                       SLD4.58
C     COMPUTE ACCELERATION                                              SLD4.59
C                                                                       SLD4.60
      COSN=COS(ANGLE)                                                   SLD4.61
      SINN=SIN(ANGLE)                                                   SLD4.62
      TERM1=G*VOL(I)*COSN*(SINN-COSN*FRIC)                              SLD4.63
      TERM2=VW*G*COSN*(COSN*FRIC-SINN)                                  SLD4.64
      TERM2=RHO*TERM2                                                   SLD4.65
      TERM3=CDV*DS*WSL*(VEL/COSN)**2                                    SLD4.66
      TERM3=RHO*TERM3                                                   SLD4.67
      A=A+TERM1+TERM2-TERM3                                             SLD4.68
   20 CONTINUE                                                          SLD4.69
```

```
      TERM4=CDP*WSL*HMAX*VEL*VEL                                        SLD4.70
      VOLUME=VTOTAL+AK*SGSL*VWTOT                                       SLD4.71
      A=(A-SGSL*TERM4)/VOLUME                                           SLD4.72
C                                                                       SLD4.73
C     TEST FOR FIRST TIME THROUGH LOOP                                  SLD4.74
C                                                                       SLD4.75
      IF (IKS .EQ. 1) GOTO 10                                           SLD4.76
      AA=AOLD+A                                                         SLD4.77
      TERM5=VEL*VEL+DS*AA                                               SLD4.78
      IF(TERM5.LT.0.) GO TO 30                                          SLD4.79
      DTSL=2.*(SQRT(TERM5)-VEL)/AA                                      SLD4.80
      VEL=VEL+DTSL*AA/2.                                                SLD4.81
      VSL(IKS,1)=VEL                                                    SLD4.82
      IF(VEL.LT.0.) GO TO 30                                            SLD4.83
      TSL(IKS)=TSL(IKS-1)+DTSL                                          SLD4.84
   10 CONTINUE                                                          SLD4.85
   30 NTV=IKS-1                                                         SLD4.86
      RETURN                                                            SLD4.87
      END                                                               SLD4.88
```

```
      SUBROUTINE VSORT(A,II,JJ)                                         VSORT.2
C                                                                       VSORT.3
C ***** CAUTION *****                                                   VSORT.4
C ***** CAUTION ***** THIS ROUTINE WILL NOT WORK PROPERLY AT HIGHEST    VSORT.5
C ***** CAUTION ***** OPTIMIZATION LEVEL OF SOME COMPILERS              VSORT.6
C ***** CAUTION *****                                                   VSORT.7
C                                                                       VSORT.8
C     ASCENDING SORT OF A VECTOR                                        VSORT.9
C                                                                       VSORT.10
C     EXTERNALS REFERENCED                                              VSORT.11
C          (NONE)                                                       VSORT.12
C     EXTERNALS CONTAINING REFERENCES                                   VSORT.13
C          LANDSLD                                                      VSORT.14
C                                                                       VSORT.15
C     FORMAL PARAMETERS                                                 VSORT.16
C       A   -  VECTOR TO BE SORTED                                      VSORT.17
C       II  -  FIRST ELEMENT TO INCLUDE IN SORT                         VSORT.18
C       JJ  -  LAST ELEMENT TO INCLUDE IN SORT                          VSORT.19
C                                                                       VSORT.20
C     MINIMUM DIMENSIONS                                                VSORT.21
C       IU,IL  -  (INT(ALOG(JJ-II+1)/ALOG(2))+1)                        VSORT.22
C                                                                       VSORT.23
C     VECTOR A IS CONSIDERED TO BE OF TYPE INTEGER REGARDLESS OF        VSORT.24
C       ITS ACTUAL TYPE                                                 VSORT.25
C                                                                       VSORT.26
      INTEGER A(1)                                                      VSORT.27
      INTEGER T,TT,IU(30),IL(30)                                        VSORT.28
      M=1                                                               VSORT.29
      I=II                                                              VSORT.30
      J=JJ                                                              VSORT.31
    5 IF (I .GE. J) GOTO 70                                             VSORT.32
   10 K=I                                                               VSORT.33
      IJ=(J+I)/2                                                        VSORT.34
      T=A(IJ)                                                           VSORT.35
      IF (A(I) .LE. T) GOTO 20                                          VSORT.36
      A(IJ)=A(I)                                                        VSORT.37
      A(I)=T                                                            VSORT.38
      T=A(IJ)                                                           VSORT.39
```

```
20 L=J                                                    VSORT.40
   IF (A(J) .GE. T) GOTO 40                               VSORT.41
   A(IJ)=A(J)                                             VSORT.42
   A(J)=T                                                 VSORT.43
   T=A(IJ)                                                VSORT.44
   IF (A(I) .LE. T) GOTO 40                               VSORT.45
   A(IJ)=A(I)                                             VSORT.46
   A(I)=T                                                 VSORT.47
   T=A(IJ)                                                VSORT.48
   GOTO 40                                                VSORT.49
30 A(L)=A(K)                                              VSORT.50
   A(K)=TT                                                VSORT.51
40 L=L-1                                                  VSORT.52
   IF (A(L) .GT. T) GOTO 40                               VSORT.53
   TT=A(L)                                                VSORT.54
50 K=K+1                                                  VSORT.55
   IF (A(K) .LT. T) GOTO 50                               VSORT.56
   IF (K .LE. L) GOTO 30                                  VSORT.57
   IF (L-I .LE. J-K) GOTO 60                              VSORT.58
   IL(M)=I                                                VSORT.59
   IU(M)=L                                                VSORT.60
   I=K                                                    VSORT.61
   M=M+1                                                  VSORT.62
   GOTO 80                                                VSORT.63
60 IL(M)=K                                                VSORT.64
   IU(M)=J                                                VSORT.65
   J=L                                                    VSORT.66
   M=M+1                                                  VSORT.67
   GOTO 80                                                VSORT.68
70 M=M-1                                                  VSORT.69
   IF (M .EQ. 0) RETURN                                   VSORT.70
   I=IL(M)                                                VSORT.71
   J=IU(M)                                                VSORT.72
80 IF(J-I .GE. 11) GOTO 10                                VSORT.73
   IF (I .EQ. II) GOTO 5                                  VSORT.74
   I=I-1                                                  VSORT.75
90 I=I+1                                                  VSORT.76
   IF (I .EQ. J) GOTO 70                                  VSORT.77
```

```
      T=A(I+1)                                                          VSORT.78
      IF(A(I) .LE. T) GOTO 90                                           VSORT.79
      K=I                                                               VSORT.80
  100 A(K+1)=A(K)                                                       VSORT.81
      K=K-1                                                             VSORT.82
      IF (T .LT. A(K)) GOTO 100                                         VSORT.83
      A(K+1)=T                                                          VSORT.84
      GOTO 90                                                           VSORT.85
      END                                                               VSORT.86
```

```
      SUBROUTINE CSORT(A,NR,KC,J1,J2,K1,K2)                             CSORT.2
C                                                                       CSORT.3
C ***** CAUTION *****                                                   CSORT.4
C ***** CAUTION ***** THIS ROUTINE WILL NOT WORK PROPERLY AT HIGHEST    CSORT.5
C ***** CAUTION ***** OPTIMIZATION LEVEL OF SOME COMPILERS              CSORT.6
C ***** CAUTION *****                                                   CSORT.7
C                                                                       CSORT.8
C     ASCENDING SORT OF AN ARRAY COLUMN (SECOND SUBSCRIPT)              CSORT.9
C                                                                       CSORT.10
C     EXTERNALS REFERENCED                                              CSORT.11
C          (NONE)                                                       CSORT.12
C     EXTERNALS CONTAINING REFERENCES                                   CSORT.13
C          SLD1                                                         CSORT.14
C                                                                       CSORT.15
C     A   ARRAY TO BE SORTED                                            CSORT.16
C     NR  DIMENSIONED NUMBER OF ROWS                                    CSORT.17
C     KC  SORT KEY COLUMN                                               CSORT.18
C     J1,J2  SORT IS OF ALL ELEMENTS IN COLUMN KC BETWEEN ROWS          CSORT.19
C            J1 AND J2 (INCLUSIVE).                                     CSORT.20
C     K1,K2  COLUMNS K1 THROUGH K2 ARE MANIPULATED THE SAME AS          CSORT.21
C            COLUMN KC.  (K1 .LE. KC .LE. K2)                           CSORT.22
C                                                                       CSORT.23
C     MINIMUM DIMENSIONS                                                CSORT.24
C       TT,SS  -  (K2)                                                  CSORT.25
C       IU,IL  -  (INT(ALOG(J2-J1+1)/ALOG(2))+1)                        CSORT.26
C                                                                       CSORT.27
C     ARRAY A IS CONSIDERED TO BE OF TYPE INTEGER REGARDLESS OF         CSORT.28
C       ITS ACTUAL TYPE.                                                CSORT.29
C                                                                       CSORT.30
      INTEGER A(NR,1)                                                   CSORT.31
      INTEGER TT(10),SS(10),IU(30),IL(30)                               CSORT.32
      INTEGER C,T                                                       CSORT.33
      M=1                                                               CSORT.34
      I=J1                                                              CSORT.35
      J=J2                                                              CSORT.36
    5 IF (I .GE. J) GOTO 95                                             CSORT.37
   10 K=I                                                               CSORT.38
      I2=(J+I)/2                                                        CSORT.39
```

```
      T=A(I2,KC)                                                        CSORT.40
      IF (A(I,KC) .LE. T) GOTO 20                                       CSORT.41
        DO 15 C=K1,K2                                                   CSORT.42
        T=A(I2,C)                                                       CSORT.43
        A(I2,C)=A(I,C)                                                  CSORT.44
15      A(I,C)=T                                                        CSORT.45
      T=A(I2,KC)                                                        CSORT.46
20    L=J                                                               CSORT.47
      IF (A(J,KC) .GE. T) GOTO 35                                       CSORT.48
        DO 25 C=K1,K2                                                   CSORT.49
        T=A(I2,C)                                                       CSORT.50
        A(I2,C)=A(J,C)                                                  CSORT.51
25      A(J,C)=T                                                        CSORT.52
      T=A(I2,KC)                                                        CSORT.53
      IF (A(I,KC) .LE. T) GOTO 35                                       CSORT.54
        DO 30 C=K1,K2                                                   CSORT.55
        T=A(I2,C)                                                       CSORT.56
        A(I2,C)=A(I,C)                                                  CSORT.57
30      A(I,C)=T                                                        CSORT.58
      T=A(I2,KC)                                                        CSORT.59
35    L=L-1                                                             CSORT.60
      IF (A(L,KC) .GT. T) GOTO 35                                       CSORT.61
        DO 40 C=K1,K2                                                   CSORT.62
40      SS(C)=A(L,C)                                                    CSORT.63
45    K=K+1                                                             CSORT.64
      IF (A(K,KC) .LT. T) GOTO 45                                       CSORT.65
      IF (K .GT. L) GOTO 55                                             CSORT.66
        DO 50 C=K1,K2                                                   CSORT.67
        A(L,C)=A(K,C)                                                   CSORT.68
50      A(K,C)=SS(C)                                                    CSORT.69
      GOTO 35                                                           CSORT.70
55    IF (L-I .LE. J-K) GOTO 60                                         CSORT.71
      IL(M)=I                                                           CSORT.72
      IU(M)=L                                                           CSORT.73
      I=K                                                               CSORT.74
      M=M+1                                                             CSORT.75
      GOTO 65                                                           CSORT.76
60    IL(M)=K                                                           CSORT.77
```

```
      IU(M)=J                                                           CSORT.78
      J=L                                                               CSORT.79
      M=M+1                                                             CSORT.80
   65 IF (J-I .GE. 11) GOTO 10                                          CSORT.81
      IF (I .EQ. J1) GOTO 5                                             CSORT.82
      I=I-1                                                             CSORT.83
   70 I=I+1                                                             CSORT.84
      IF (I .EQ. J) GOTO 95                                             CSORT.85
      T=A(I+1,KC)                                                       CSORT.86
      IF (A(I,KC) .LE. T) GOTO 70                                       CSORT.87
      K=I                                                               CSORT.88
        DO 75 C=K1,K2                                                   CSORT.89
   75   TT(C)=A(I+1,C)                                                  CSORT.90
   80   DO 85 C=K1,K2                                                   CSORT.91
   85   A(K+1,C)=A(K,C)                                                 CSORT.92
      K=K-1                                                             CSORT.93
      IF (T .LT. A(K,KC)) GOTO 80                                       CSORT.94
        DO 90 C=K1,K2                                                   CSORT.95
   90   A(K+1,C)=TT(C)                                                  CSORT.96
      GOTO 70                                                           CSORT.97
   95 M=M-1                                                             CSORT.98
      IF (M .EQ. 0) RETURN                                              CSORT.99
      I=IL(M)                                                           CSORT.100
      J=IU(M)                                                           CSORT.101
      GOTO 65                                                           CSORT.102
      END                                                               CSORT.103
```

```
      SUBROUTINE ERR(N,ILEV,NM,ERRM,NPP,                                ERR.2
     A  A11,A21,A31,A12,A22,A32,A13,A23,A33,A14,A24,A34,A15,A25,A35,    ERR.3
     B  A16,A26,A36,A17,A27,A37,A18,A28,A38,A19,A29,A39,A10,A20,A30)    ERR.4
C                                                                       ERR.5
C     PRINT ERROR MESSAGE, TERMINATE PROGRAM IF ILEV=1 OR -1            ERR.6
C                                                                       ERR.7
C     EXTERNALS REFERENCED                                              ERR.8
C          ABORT   (CDC LIBRARY)                                        ERR.9
C     EXTERNALS CONTAINING REFERENCES                                   ERR.10
C          DATAROR                                                      ERR.11
C          DATAPRT                                                      ERR.12
C          INITIAL                                                      ERR.13
C          INITFOR                                                      ERR.14
C          INITNAT                                                      ERR.15
C          SLD1                                                         ERR.16
C          SLD3                                                         ERR.17
C                                                                       ERR.18
      IMPLICIT INTEGER (A-Z)                                            ERR.19
      DIMENSION FMT(2),IFMT(4)                                          ERR.20
      DIMENSION I(10),H(10),Z(10)                                       ERR.21
      DIMENSION ERRM(1)                                                 ERR.22
      DIMENSION IEM(11,7),IC(6,7),IN(10,7)                              ERR.23
      DIMENSION INC(10)                                                 ERR.24
      DIMENSION NERRS(100)                                              ERR.25
      DATA INC/1,4,4,3,4,6,2,3,5,0/                                     ERR.26
      DATA INN/9/                                                       ERR.27
      DATA IC/                                                          ERR.28
     A #XXXXXXXXX #,#XXX    XX #,#   XXXX   #,#XXXXXXXXX #,#   #,# #,   ERR.29
     B #XX        #,#XXXX   XX #,# XX    XX #,#XX      XX#,#   #,# #,   ERR.30
     C #XX        #,#XXXXX  XX #,#XX      XX#,#XX      XX#,#   #,# #,   ERR.31
     D #XXXXXX    #,#XX XXX XX #,#XX      XX#,#XXXXXXXXX #,#   #,# #,   ERR.32
     E #XX        #,#XX  XXXXX #,#XX      XX#,#XX  XXX   #,#   #,# #,   ERR.33
     F #XX        #,#XX   XXXX #,# XX    XX #,#XX   XXX  #,#XXX#,# #,   ERR.34
     G #XXXXXXXXX #,#XX    XXX #,#   XXXX   #,#XX    XXX #,#XXX#,# #/   ERR.35
      DATA IN/                                                          ERR.36
     A #  XXXXXX  #,#      XX  #,# XXXXXXXX #,#XXXXXXXX #,#   XXX    #,ERR.37
     A #XXXXXXXXX #,#   XXXX   #,#XXXXXXXXXX#,#  XXXXXX  #,# XXXXXXXX #,ERR.38
     B # XX    XX #,#     XXX  #,#XXX     XX#,#     XXX  #,#  XXX  XX #,ERR.39
```

```
B ≠XX          ≠,≠ XXX       ≠,≠        XXX≠,≠ XX     XX ≠,≠XX       XX≠,ERR.40
C ≠XX       XX≠,≠     XXXX  ≠,≠         XX≠,≠     XXX    ≠,≠ XXX   XX ≠,ERR.41
C ≠XXXXXXXX  ≠,≠XX          ≠,≠       XXX ≠,≠  XXXXXX  ≠,≠XX       XX≠,ERR.42
D ≠XX       XX≠,≠       XX  ≠,≠       XXX ≠,≠       XXX ≠,≠XXX    XX ≠,ERR.43
D ≠       XXX ≠,≠XXXXXXXXX ≠,≠      XXX  ≠,≠ XX     XX ≠,≠ XXXXXXXXX≠,ERR.44
E ≠XX       XX≠,≠       XX  ≠,≠    XXX    ≠,≠        XXX≠,≠XXXXXXXXXX≠,ERR.45
E ≠        XXX≠,≠XX       XX≠,≠    XXX    ≠,≠XX       XX≠,≠       XXX ≠,ERR.46
F ≠ XX     XX ≠,≠       XX  ≠,≠  XXX      ≠,≠XXX    XXX ≠,≠        XX ≠,ERR.47
F ≠XXX    XXX ≠,≠XX       XX≠,≠   XXX     ≠,≠XXX     XXX≠,≠      XXX  ≠,ERR.48
G ≠  XXXXXX  ≠,≠     XXXXXX≠,≠XXXXXXXXXX≠,≠ XXXXXXX  ≠,≠        XX ≠,ERR.49
G ≠ XXXXXXX  ≠,≠ XXXXXXXX ≠,≠  XXX      ≠,≠  XXXXXX  ≠,≠     XXX    ≠/ERR.50
      DATA NERRS/100*0/                                                ERR.51
      DATA IFMT/10H1H=,G16.6),10H1H=,I12)  ,10H1H=,F12.3),1H /         ERR.52
      DATA FMT/10H(4XA10,1X ,1H /                                      ERR.53
C                                                                      ERR.54
      NERRS(100)=NERRS(100)+1                                          ERR.55
      NERRS(N)=NERRS(N)+1                                              ERR.56
      IF (ILEV .EQ. 0) GOTO 35                                         ERR.57
      DO 10 J=1,7                                                      ERR.58
      DO 10 K=1,INN                                                    ERR.59
   10 IEM(K,J)=IC(INC(K),J)                                            ERR.60
      IERR=N                                                           ERR.61
      IP=10                                                            ERR.62
      IA=0                                                             ERR.63
      K=INN                                                            ERR.64
      DO 20 II=1,2                                                     ERR.65
      IA=IERR/IP                                                       ERR.66
      IERR=IERR-IP*IA                                                  ERR.67
      IP=IP/10                                                         ERR.68
      K=K+1                                                            ERR.69
      DO 20 J=1,7                                                      ERR.70
      IEM(K,J)=IN(IA+1,J)                                              ERR.71
   20 CONTINUE                                                         ERR.72
      IF (N .GT. 9) GOTO 30                                            ERR.73
      DO 25 J=1,7                                                      ERR.74
   25 IEM(INN+1,J)=1H                                                  ERR.75
   30 CONTINUE                                                         ERR.76
      INN2=INN+2                                                       ERR.77
```

```
      PRINT 101,((IEM(K,J),K=1,INN2),J=1,7)                              ERR.78
      GOTO 40                                                            ERR.79
   35 PRINT 102,N                                                        ERR.80
   40 CONTINUE                                                           ERR.81
      IF (NM .EQ. 0) GOTO 50                                             ERR.82
      PRINT 104                                                          ERR.83
      PRINT 103,(ERRM(K),K=1,NM)                                         ERR.84
   50 CONTINUE                                                           ERR.85
      NP=MIN0(10,MAX0(0,NPP))                                            ERR.86
      IF (NP .EQ. 0) GOTO 70                                             ERR.87
      I(1)=A11 $ H(1)=A21 $ Z(1)=A31 $ IF (NP .EQ. 1) GOTO 1             ERR.88
      I(2)=A12 $ H(2)=A22 $ Z(2)=A32 $ IF (NP .EQ. 2) GOTO 1             ERR.89
      I(3)=A13 $ H(3)=A23 $ Z(3)=A33 $ IF (NP .EQ. 3) GOTO 1             ERR.90
      I(4)=A14 $ H(4)=A24 $ Z(4)=A34 $ IF (NP .EQ. 4) GOTO 1             ERR.91
      I(5)=A15 $ H(5)=A25 $ Z(5)=A35 $ IF (NP .EQ. 5) GOTO 1             ERR.92
      I(6)=A16 $ H(6)=A26 $ Z(6)=A36 $ IF (NP .EQ. 6) GOTO 1             ERR.93
      I(7)=A17 $ H(7)=A27 $ Z(7)=A37 $ IF (NP .EQ. 7) GOTO 1             ERR.94
      I(8)=A18 $ H(8)=A28 $ Z(8)=A38 $ IF (NP .EQ. 8) GOTO 1             ERR.95
      I(9)=A19 $ H(9)=A29 $ Z(9)=A39 $ IF (NP .EQ. 9) GOTO 1             ERR.96
      I(10)=A10 $ H(10)=A20 $ Z(10)=A30                                  ERR.97
    1 CONTINUE                                                           ERR.98
      PRINT 104                                                          ERR.99
      DO 60 K=1,NP                                                       ERR.100
      FMT(2)=IFMT(I(K))                                                  ERR.101
      PRINT FMT,H(K),Z(K)                                                ERR.102
   60 CONTINUE                                                           ERR.103
   70 CONTINUE                                                           ERR.104
      IF (ILEV) 75,80,85                                                 ERR.105
   75 CALL ABORT                                                         ERR.106
   80 RETURN                                                             ERR.107
   85 STOP                                                               ERR.108
C                                                                        ERR.109
  101 FORMAT (1H1,2(/1X130(1H/))///7(/1X10(A10,2X),A10)///               ERR.110
     A      2(/1X130(1H/))///)                                           ERR.111
  102 FORMAT (1H0,23H**********   ERROR NO. ,I2,13H   **********)        ERR.112
  103 FORMAT (3X,8A10)                                                   ERR.113
  104 FORMAT (1H )                                                       ERR.114
      END                                                                ERR.115
```

B9.0 SAMPLE INPUT AND OUTPUT

B9.1 Listing of Sample Input Data

INPUT DECK: SLA91

```
 CARD                1         2         3         4         5         6         7         8     CARD
NUMBER    12345678901234567890123456789012345678901234567890123456789012345678901234567890    NUMBER
          --------------------------------------------------------------------------------

/  1/     SLA91                                                                                /  1/
/  2/        14   64    0    1    0    1    1    1                                           /  2/
/  3/     0.        300.      0.687795  250.      250.      1.        7163.                   /  3/
/  4/     0.025     0.7       4.0       32.17                                                 /  4/
/  5/        13    3    1    2    1    0    0    1    1    2    1    1    1    1              /  5/
/  6/     100.      0.                                                                        /  6/
/  7/                                                                                         /  7/
/  8/                                                                                         /  8/
/  9/                                                                                         /  9/
/ 10/                                                                                         / 10/

/ 11/                                                                                         / 11/
/ 12/                                                                                         / 12/
/ 13/                                                                                         / 13/
/ 14/                                        7130 7060 6980 6930 7090 7120                    / 14/
/ 15/                                   7140      7160 7160 7160 7160 7160                    / 15/
/ 16/          7150                                                                           / 16/
/ 17/                                                  7090 7050 7080 7120 7150      7160     / 17/
/ 18/                                             6990 6900 6900 6900 6930 7140               / 18/
/ 19/          6990 6940 7000 6940 6960 6900 6960 7026 7026 7026 7026 7026 7080 7120 7100     / 19/
/ 20/          7070                                                   7140                    / 20/

/ 21/                         7150 7160 7160      7090 6990 6980 6930 6920 6930 6960 7060     / 21/
/ 22/     7095 7129 7156 7140 7160      6950 6890 6910 7020 7130 7040 6960 6910 6920          / 22/
/ 23/          6780 6790 6770 6780 6790 6780 6790 6888 6888 6888 6888 6888 6920 7000 6980     / 23/
/ 24/     7080 7020 7100                      7010      7010 7060 7080 7060 7060               / 24/
/ 25/                    7110 7040 7070 7050 7140 6960 6920 6910 6880 6870 6860 6880 6922     / 25/
/ 26/     6953 6984 7017 7020 7000 6900 6890 6960 7110                      7100 6960          / 26/
/ 27/          6920 6980 6970 6920 6990 6990 7040 7000 6960 6960 6870 6850 6810 6810 6830     / 27/
/ 28/     6960 6980 7010           7140      7020 6961 6940 6890 6880 6900 6930 7030 7100     / 28/
/ 29/     7030 7010 7050 7060 7020 6900 6880 6860 6860 6876 6880 6960 7000 7000 6980 6940     / 29/
/ 30/     6920 6890 6882 6890 6890 6910 7000 7140                                             / 30/

/ 31/                                                 7130 7120 7060 6960 6910 6840           / 31/
/ 32/     6840 6880 6950 7130      7060 7040 6960 6860 6850 6830 6830 6840 6840 6860 6870     / 32/
/ 33/     6870 6960 6870 6880 6860 6860 6899 6940 6976 7013 7080                              / 33/
/ 34/     7080 7050 6970 6960 6970 7050 7130                                                  / 34/
/ 35/                                                                      7120 7010          / 35/
/ 36/     6850 6820 6880 7080 7070 6970 6870 6840 6840 6960 7060 7040 6970 6880 6890 6900     / 36/
/ 37/     6900 6910 6880 6880 6900 6950 7033 7070 7196 7141                                   / 37/
/ 38/                7140                                                                     / 38/
/ 39/                                                                            7070         / 39/
/ 40/     6940 6840 6840 6940 6970 6860 6830 6910 7080                 7080 7040 7030 7030     / 40/

          --------------------------------------------------------------------------------
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
                   1         2         3         4         5         6         7         8
```

INPUT DECK: SLA91

```
 CARD                1         2         3         4         5         6         7         8     CARD
NUMBER     12345678901234567890123456789012345678901234567890123456789012345678901234567890    NUMBER
           --------------------------------------------------------------------------------
/ 41/       7040 7040 7000      7070      7162                                                 / 41/
/ 42/                                                                                          / 42/
/ 43/                                                                                 7160     / 43/
/ 44/       7020 6960 6840 6840 6840 6830 6910 7000                                            / 44/
/ 45/                 7160                                                                     / 45/
/ 46/                                                                                          / 46/
/ 47/                                                                                          / 47/
/ 48/       7150 7030 6990 6870 6870 6960 7080                                                 / 48/
/ 49/                                                                                          / 49/
/ 50/                                                                                          / 50/

/ 51/                                                                                          / 51/
/ 52/                 7100 7000 7040 7160                                                      / 52/
/ 53/                                                                                          / 53/
/ 54/                                                                                          / 54/
/ 55/                                                                                          / 55/
/ 56/                      7100                                                                / 56/
/ 57/                                                                                          / 57/
/ 58/                                                                                          / 58/
/ 59/                                                                                          / 59/
/ 60/                                                                                          / 60/

/ 61/                                                                                          / 61/
/ 62/                                                                                          / 62/
/ 63/       63  4  5                                                                           / 63/
/ 64/        2  4  6 5.75                                                                      / 64/
/ 65/          4    2    5    2    6    2    5    8    6   12    7   17    9   20   11   20    / 65/
/ 66/          7   28    7   36    5   44    5   52    3   60                                  / 66/
/ 67/       11.        40.                                                                     / 67/
/ 68/       2          SLA91                                                                   / 68/
/ 69/          1    2   11                                                                     / 69/
/ 70/       0.         1250.     3.90      8.5                                                / 70/

/ 71/       15.81139   13.82895                                                                / 71/
/ 72/       0.         722.                                                                    / 72/
/ 73/       400.       0.                                                                      / 73/
/ 74/       0.         .016      .103      .190      .277      .364      .451      .538        / 74/
/ 75/       .625       .712      .799                                                          / 75/
/ 76/       0.         .24       1.54      2.48      3.37      3.79      3.95      3.66        / 76/
/ 77/       3.34       2.72      0.                                                            / 77/
/ 78/       END OF DATA - SLA91                                                                / 78/

           --------------------------------------------------------------------------------
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
                     1         2         3         4         5         6         7         8
```

B9.2 Description of Input Data

The input data for a sample run "SLA91" as listed in Section B9.1, are explained in this section according to the input sequence. Refer to Section B6.0 for data group number, input format, variable names, and the description of the variable. Refer to Section A3.1 in Part A for the description of computer run for "SLA91". Some self-explanatory variables are not discussed.

For Group 1 of the input cards, 1 card is used to input the title for this sample run. The title will be centered when printed. For Group 2, M and N are 14 and 64, respectively, according to Figure A7 in Part A. There is only one dam segment and one open-boundary segment normal to the y-axis (Figure A7). Therefore, MOB, NOB, MWEIR, and NWEIR are 0, 1, 0, 1, respectively. Variables IC and ICC are both 1 to use the full nonlinear model. In case of the occurrence of nonlinear instability, one or both of these two flags may be set to zero in order to improve the numerical stability.

For Group 3, TZERO and TMAX are self-explanatory. DT is 0.687795 sec which is half of the time step used in the laboratory to measure the slide motion and wave heights. DT was originally 0.087 x 15.81 where 0.087 is the model time and 15.81 is the conversion factor to obtain the proto time. This original time step was halved to check the sensitivity of the results with respect to the computational time step. The new DT of 0.687795 was not changed back even though the results are essentially the same between two preliminary runs. DX and DY are both 250 ft (see Figure B1). When they were set to 125 ft instead, the results are not too much different. When the grid spacings are 250 ft and the maximum water depth is 393 ft, the basic Courant criterion requires that DT be smaller than or equal to DX*DY/SQRT((DX*DX+DY*DY) $*G*D_{max}$)=1.57 sec. If the wave length is considered, the

factor $(1+E_{max}/H_{max})$, where E_{max} and H_{max} are the maximum wave amplitude and the maximum total water depth, respectively, should be considered. Note that $H_{max} = D_{max}+E_{max}$ are the values at the same grid point. These two values are not available until the computation is completed. Before the computation, some guess must be made. After computation, the guess can be checked. At point (5,4), where D = 393 ft is the maximum depth in the basin, E_{max} is 53 ft as can be seen from the table of maximum surface elevation in the computer output. The factor is (1+53/(393+53)) = 1.12. Therefore, at this grid point, DT should not exceed 1.57/1.12 = 1.40 sec. At (7,11), where E_{max} = 260 ft is the highest maximum in that table, the depth is 33, and the factor is therefore (1+260/(33+260)) = 1.89. Therefore, at this point, DT should not exceed 250x250/SQRT((250x250+250x250)x32.17x33)/ 1.89 = 2.87 sec. Theoretically, the checking procedure should be applied to every grid point. Practically, depending on experience, only a couple of points with deep water need checking. A safety factor of 1.1 to 2.0 may be given to insure stability. The computational cost is approximately proportional to the inverse of DT. Note that an extremely small DT usually does not improve the nonlinear instability which is still a topic for research.

The variable DCONV in Group 3 is set to one. This variable is useful when water depth is measured in fathoms or some other units different from the adopted system. Initial surface elevation is 7163 ft for the run "SLA91."

In Group 4, the Manning coefficient is 0.025 which probably should be about 0.040 to 0.060 if the ratio of the coefficients, between the model value and the prototype value, of 1:2.51, as mentioned in Section A3.1.1 of Part A, applies. However, the propagation of a wave in relatively deep water is insensitive to the bottom friction. The results should be very close when the high bound and the low bound of the coefficient values are used.

The energy dissipation coefficient, CED, is discussed in Section A2.1.1 (Figure A2) of Part A and Section B2.4.3 (Figures B6 and B7) of Part B. This variable has a constant value applied to the whole partial reflective boundary. This coefficient can be calibrated against some particular results of interest. The discharge coefficient, CD, is set to 4.0 here (see Section B2.5). Figure B16 shows that the wave height at Morrow Point Dam is not sensitive to the discharge coefficient. Four series of computer runs, A, B, C and D, in Figure B16 are all for "SLA91." They are different by slight changes of landslide velocity and energy dissipation coefficient.

Groups 5 and 6 contain flags and parameters for the control of computer output. Group 7 has an array for either water depth or surface elevation, depending on whether Z0 in Group 3 is zero or not. There are M sets of cards. Each set has N values.

Groups 8 to 11 specify the location of the open boundaries and dams, if any. Refer to Sections B2.4.1 and B2.4.2 if necessary. Groups 12 to 14 are self-explanatory. the "forced" landslide submodel is used for this sample run. The sub-title in Group 14 is not centered in the output.

Groups 15 to 23 define the wave-generating parameters. In Group 15, NSL is 1 and NXH is 2 for the landslides in Morrow Point Reservoir because the landslide in the laboratory is modeled by a simple reversed wedge. NTV is obtained from the laboratory data. The default of 50 is used for both MSUB and NSUB (see Figure B11 for sensitivity test).

In Group 16, the forward direction of the landslide, THETA, is 0, because the landslide moves along the direction of the x-axis. The initial position of the landslide is defined by XO and YO which specify the location of the right corner of the leading edge of the landslide. The value of XO is

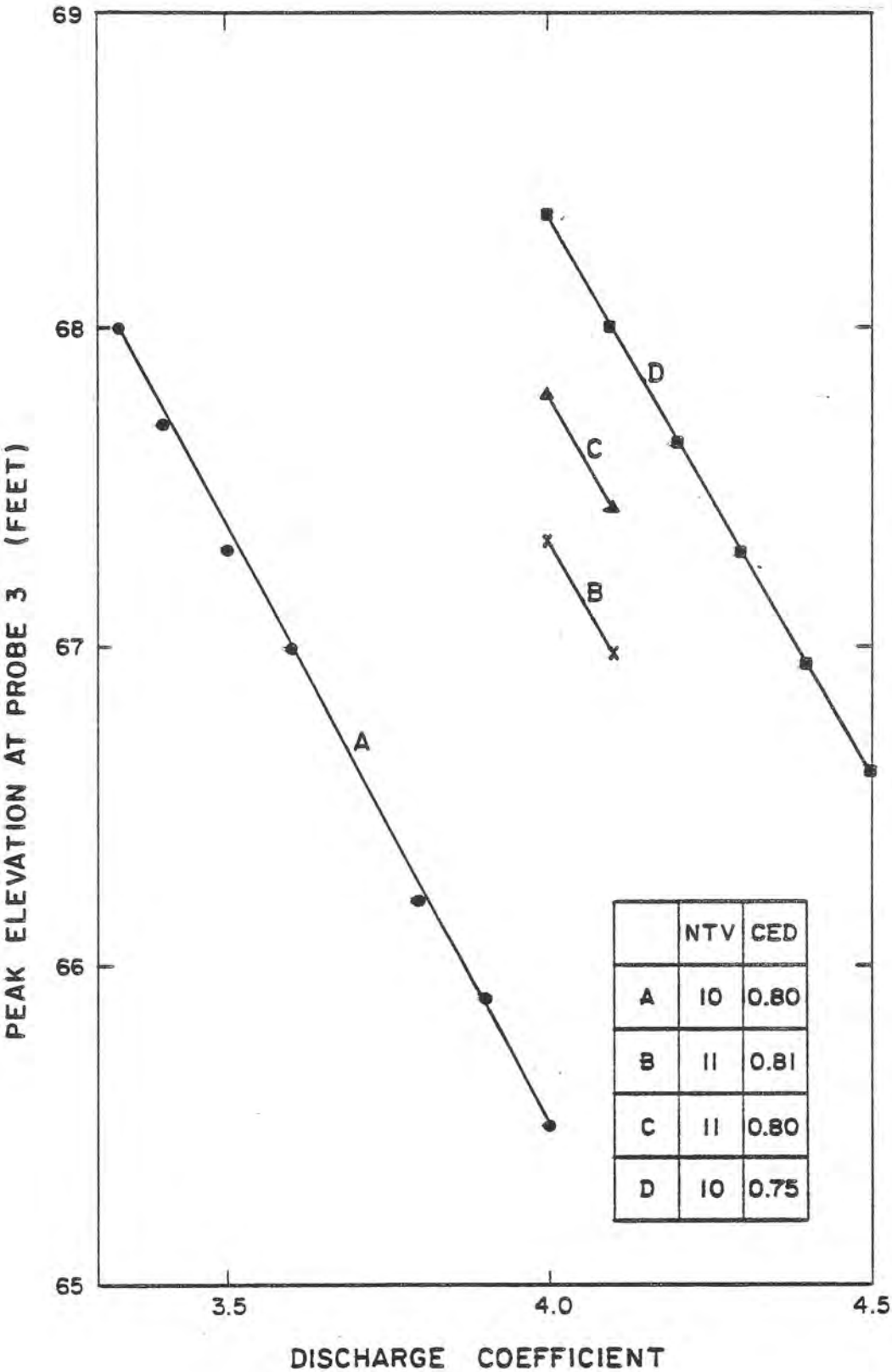


FIGURE B16: Sensitivity test of the discharge coefficient in the weir formula.

adjusted to 3.9 such that the initial submerged volume of the landslide is the same as the laboratory data of 1.16x 250^3 cubic feet. The volume per unit width is $1.16 \times 250^3/1250 = 14500$ ft^2. Therefore, the horizontal distance of the submerged landslide is obtained from solving for S from

$$\frac{1}{2} S(S \tan 29°) = 14500 \qquad \text{(B10)}$$

where 29° is the inclination of the sliding slope. The calculated S is 228.7 ft which is equivalent to 228.7/250 = 0.91 coordinate units. Since the land-water boundary in the numerical model is located at i = 2.0, the x-coordinate of the leading edge of the initial landslide position is XO = 2.0+0.91 = 3.9.

In Group 17, conversion factors are used to convert the model values to prototype values. Original landslide velocity was measured along the slope. The conversion factor to convert this velocity to the horizontal velocity is included in VSLCNV. Groups 18 and 19 are skipped because LSKEY is 2. Groups 20 to 23 describe the spatial distribution and the velocity of the landslide. These data are obtained from the laboratory data.

B9.3 Listing of Sample Output

Part of the printed output is listed in the following pages. This set of output was obtained by running LSWAVE as listed in Section B8.0 with input data as listed in Section B9.1. This is a partial listing. The page numbers based on the full listing are typed on the right upper corner of each page.

```
-- LSWAVE --                          L      SSS  W   W   A   V   V EEEEE                          -- DEVELOPED BY --
VERSION  A.3                          L     S     W   W  A A  V   V E                              TETRA TECH, INC.
                                      L      SSS  W W W A   A V   V EEE                            630  N. ROSEMEAD BLVD.
11.43.28   01 DEC 81                  L         S W W W AAAAA  V V  E                              PASADENA, CALIF. 91107
                                      LLLLL SSSS   W W  A   A   V   EEEEE                             (213) 449-6400
```

```
-----------------
---   SLA91   ---
-----------------
```

VARIABLE	VALUE	DESCRIPTION OR => RESULT
M	14	TOTAL NUMBER OF GRID POINTS IN X-DIRECTION
N	64	TOTAL NUMBER OF GRID POINTS IN Y-DIRECTION
MOB	0	NUMBER OF OPEN BOUNDARY SEGMENTS NORMAL TO X-AXIS
NOB	1	NUMBER OF OPEN BOUNDARY SEGMENTS NORMAL TO Y-AXIS
MWEIR	0	NUMBER OF DAM SEGMENTS NORMAL TO X-AXIS
NWEIR	1	NUMBER OF DAM SEGMENTS NORMAL TO Y-AXIS
IC	1	=> CONVECTIVE TERM INCLUDED
ICC	1	=> CONTINUITY EQUATION NON-LINEAR
TZERO	0.	BEGINNING TIME (SECONDS)
TMAX	300.000	ENDING TIME (SECONDS)
DT	.687795	TIME STEP (SECONDS)
DX	250.000	GRID SPACING IN X-DIRECTION
DY	250.000	GRID SPACING IN Y-DIRECTION
DCONV	1.00000	CONVERSION FACTOR FOR DEPTHS
Z0	7163.00	INITIAL SURFACE ELEVATION
CHANN	.250000E-01	MANNING ROUGHNESS COEFFICIENT
CED	.700000	COEFFICIENT OF ENERGY DISSIPATION AT PARTIALLY REFLECTIVE BOUNDARY
CD	4.00000	DISCHARGE COEFFICIENT FOR WEIR FORMULA
G	32.1700	GRAVITATIONAL ACCELERATION
NES	13	NUMBER OF POINTS FOR WHICH HISTORIES ARE STORED
IS	3	NUMBER OF TIME STEPS BETWEEN STORED HISTORY VALUES
ISP	1	ISP*IS = NUMBER OF TIME STEPS BETWEEN PRINTED HISTORY VALUES
ILR(1,1)	2	=> FULL PAGE PLOTS OF SURFACE ELEVATION HISTORIES
ILR(2,1)	1	=> SURFACE ELEVATION HISTORIES PRINTED
ILR(1,2)	0	=> NO PLOTS OF VELOCITY HISTORIES
ILR(2,2)	0	=> VELOCITY HISTORIES NOT PRINTED
ILR(1,3)	1	=> HALF PAGE PLOT OF OVER-TOPPING RATE HISTORY
ILR(2,3)	1	=> OVER-TOPPING HISTORY PRINTED
KS	2	NUMBER OF SNAPSHOTS SPECIFIED IN TS(*)
LSE	1	=> SNAPSHOTS OF SURFACE ELEVATION PRINTED FOR SPECIFIED TIMES (TS(*))
LSU	1	=> SNAPSHOTS OF VELOCITY PRINTED FOR SPECIFIED TIMES (TS(*))
IENV	1	=> MAXIMUM SURFACE ELEVATION PRINTED
IPLTF	1	=> FILE WRITTEN FOR EXTERNAL PLOTS
EMAX	1[illegible].000	MAXIMUM RANGE FOR ELEVATION PLOTS
UMAX	0.	(UNUSED - NO VELOCITY PLOTS)

INPUT GRID

I \ J	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
2	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
3	-0.	-0.	-0.	-0.	-0.	-0.	7140.0	-0.	7160.0	7160.0	7160.0	7160.0	7160.0	-0.	-0.	-0.
4	-0.	6990.0	6940.0	7010.0	6940.0	6960.0	6910.0	6960.0	7026.0	7026.0	7026.0	7026.0	7026.0	7080.0	7120.0	7160.0
5	-0.	6780.0	6790.0	6770.0	6780.0	6790.0	6780.0	6790.0	6880.0	6880.0	6880.0	6880.0	6880.0	6920.0	7000.0	6980.0
6	-0.	6920.0	6980.0	6970.0	6920.0	6990.0	6990.0	7040.0	7000.0	6960.0	6960.0	6870.0	6850.0	6810.0	6830.0	6830.0
7	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	7130.0	7120.0	7060.0	6960.0	6910.0	6840.0
8	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	7120.0	7010.0
9	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	7070.0
10	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	7160.0
11	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
12	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
13	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
14	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.

INPUT GRID (CONTINUED)

I \ J	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
2	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
3	-0.	7150.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
4	-0.	7070.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	7140.0	-0.	-0.	-0.
5	7080.0	7020.0	7100.0	-0.	-0.	-0.	-0.	7010.0	-0.	7010.0	7060.0	7080.0	7060.0	7060.0	-0.	-0.
6	6960.0	6980.0	7010.0	-0.	-0.	7140.0	-0.	7020.0	6960.0	6940.0	6890.0	6880.0	6900.0	6930.0	7030.0	7100.0
7	6840.0	6880.0	6950.0	7130.0	-0.	7060.0	7040.0	6960.0	6860.0	6850.0	6830.0	6830.0	6840.0	6840.0	6860.0	6870.0
8	6850.0	6820.0	6880.0	7080.0	7070.0	6970.0	6870.0	6840.0	6840.0	6960.0	7060.0	7040.0	6970.0	6880.0	6890.0	6900.0
9	6940.0	6840.0	6840.0	6940.0	6970.0	6860.0	6830.0	6910.0	7080.0	-0.	-0.	-0.	7080.0	7040.0	7030.0	7030.0
10	7020.0	6960.0	6840.0	6840.0	6840.0	6830.0	6910.0	7000.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
11	7150.0	7030.0	6990.0	6870.0	6870.0	6960.0	7080.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
12	-0.	-0.	7100.0	7060.0	7040.0	7160.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
13	-0.	-0.	-0.	7100.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
14	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.

INPUT GRID (CONTINUED)

I \ J	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
1	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
2	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
3	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	7090.0	7050.0	7080.0	7120.0	7160.0	-0.	7160.0
4	-0.	-0.	-0.	-0.	7150.0	7160.0	7160.0	-0.	7090.0	6990.0	6980.0	6930.0	6920.0	6930.0	6960.0	7060.0
5	-0.	-0.	-0.	7110.0	7040.0	7070.0	7050.0	7040.0	6960.0	6920.0	6910.0	6880.0	6870.0	6880.0	6880.0	6922.0
6	7030.0	7010.0	7050.0	7060.0	7020.0	6990.0	6880.0	6860.0	6860.0	6876.0	6880.0	6960.0	7000.0	7000.0	6980.0	6940.0
7	6870.0	6860.0	6870.0	6880.0	6880.0	6860.0	6890.0	6940.0	6976.0	7013.0	7080.0	-0.	-0.	-0.	-0.	-0.
8	6900.0	6910.0	6880.0	6880.0	6900.0	6950.0	7033.0	7070.0	7106.0	7141.0	-0.	-0.	-0.	-0.	-0.	-0.
9	7040.0	7040.0	7000.0	-0.	7070.0	-0.	7162.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
10	-0.	-0.	7160.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
11	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
12	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
13	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
14	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.

INPUT GRID (CONTINUED)

I \ J	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
1	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
2	-0.	-0.	-0.	-0.	-0.	-0.	-0.	7130.0	7060.0	6980.0	6930.0	7090.0	7120.0	-0.	-0.	-0.
3	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	6990.0	6900.0	6500.0	6900.0	6930.0	7140.0	-0.	-0.
4	7095.0	7129.0	7156.0	7140.0	7160.0	-0.	6950.0	6850.0	6910.0	7020.0	7130.0	7040.0	6960.0	6910.0	6920.0	-0.
5	6953.0	6984.0	7017.0	7020.0	7000.0	6910.0	6890.0	6960.0	7110.0	-0.	-0.	-0.	-0.	7100.0	6960.0	-0.
6	6920.0	6890.0	6882.0	6890.0	6890.0	6910.0	7000.0	7140.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
7	7080.0	7050.0	6970.0	6960.0	6970.0	7050.0	7130.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
8	-0.	-0.	7140.0	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
9	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
10	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
11	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
12	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
13	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.
14	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.	-0.

DEPTH GRID

I \ J	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	0.	0.	0.	0.	0.	0.	23.0	0.	3.0	3.0	3.0	3.0	3.0	0.	0.	0.
4	0.	173.0	223.0	163.0	223.0	273.0	263.0	203.0	137.0	137.0	137.0	137.0	137.0	83.0	43.0	63.0
5	0.	383.0	373.0	393.0	383.0	373.0	363.0	372.0	276.0	275.0	275.0	276.0	275.0	243.0	163.0	183.0
6	0.	243.0	183.0	193.0	243.0	173.0	173.0	123.0	163.0	203.0	203.0	293.0	313.0	353.0	353.0	323.0
7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	33.0	43.0	103.0	203.0	253.0	323.0
8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	43.0	153.0
9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	93.0
10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	3.0
11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

DEPTH GRID (CONTINUED)

I \ J	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	0.	13.0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4	0.	93.0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	23.0	0.	0.	0.
5	83.0	143.0	63.0	0.	0.	0.	0.	153.0	0.	153.0	103.0	83.0	103.0	103.0	0.	0.
6	203.0	183.0	153.0	0.	0.	23.0	0.	142.0	203.0	223.0	273.0	283.0	263.0	233.0	133.0	63.0
7	323.0	283.0	213.0	33.0	0.	103.0	123.0	203.0	303.0	313.0	333.0	333.0	323.0	323.0	303.0	293.0
8	313.0	343.0	283.0	83.0	93.0	193.0	293.0	323.0	323.0	203.0	103.0	123.0	193.0	283.0	273.0	263.0
9	223.0	323.0	323.0	223.0	193.0	303.0	333.0	283.0	83.0	0.	0.	0.	83.0	123.0	133.0	133.0
10	143.0	203.0	323.0	323.0	323.0	333.0	253.0	162.0	0.	0.	0.	0.	0.	0.	0.	0.
11	13.0	133.0	173.0	293.0	293.0	203.0	83.0	0.	0.	0.	0.	0.	0.	0.	0.	0.
12	0.	0.	63.0	163.0	123.0	3.0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	0.	0.	0.	63.0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

DEPTH GRID (CONTINUED)

I \ J	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	0.	0.	0.	0.	0.	0.	0.	0.	0.	73.0	113.0	83.0	43.0	13.0	0.	3.0
4	0.	0.	0.	0.	13.0	3.0	3.0	0.	73.0	173.0	183.0	233.0	243.0	233.0	203.0	103.0
5	0.	0.	0.	53.0	123.0	93.0	113.0	123.0	203.0	243.0	253.0	283.0	293.0	283.0	283.0	241.0
6	133.0	153.0	113.0	103.0	143.0	263.0	283.0	303.0	303.0	287.0	283.0	203.0	163.0	163.0	183.0	223.0
7	293.0	303.0	293.0	283.0	283.0	303.0	264.0	223.0	187.0	150.0	83.0	0.	0.	0.	0.	0.
8	263.0	253.0	283.0	283.0	263.0	213.0	130.0	53.0	57.0	22.0	0.	0.	0.	0.	0.	0.
9	123.0	123.0	163.0	0.	93.0	0.	1.0	0.	0.	0.	0.	0.	0.	0.	0.	0.
10	0.	0.	3.0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

DEPTH GRID (CONTINUED)

I \ J	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	33.0	103.0	183.0	233.0	73.0	43.0	0.	0.	0.
3	0.	0.	0.	0.	0.	0.	0.	0.	173.0	263.0	263.0	263.0	233.0	23.0	0.	0.
4	68.0	34.0	7.0	23.0	3.0	0.	213.0	273.0	253.0	143.0	33.0	123.0	203.0	253.0	243.0	0.
5	210.0	179.0	146.0	143.0	163.0	263.0	273.0	203.0	53.0	0.	0.	0.	0.	63.0	203.0	0.
6	243.0	273.0	281.0	273.0	273.0	253.0	163.0	23.0	0.	0.	0.	0.	0.	0.	0.	0.
7	83.0	113.0	193.0	203.0	193.0	113.0	33.0	0.	0.	0.	0.	0.	0.	0.	0.	0.
8	0.	0.	23.0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

OPEN BOUNDARIES NORMAL TO Y-AXIS

K (K=1,NOB)	JOB(K) (COL)	I1OB(K) (BEG ROW)	I2OB(K) (END ROW)
1	63	4	5

DAM SEGMENTS NORMAL TO Y-AXIS

K (K=1,NWEIR)	JWEIR(K) (COL)	I1WEIR(K) (BEG ROW)	I2WEIR(K) (END ROW)	HWEIRJ(K) (HEIGHT)
1	2	4	6	5.75

BOUNDARIES PLOTTED ON COMPUTATIONAL GRID --

/ - DAM
= - OPEN BOUNDARY
+ - D(I,J) .GT. 0.
D - D(I,J) .EQ. 0.
- - D(I,J) .LT. 0.

```
      J         1         2         3         4         5         6
       123456789012345678901234567890123456789012345678901234567890123456789012345
    I
    1  DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
    2  DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
    3  DDDDDD+D+++++DDDD+DDDDDDDDDDDDDDDDDDDDDDDD+++++D+DDDDDDDDD++++++DD
    4  D/+++++++++++++D+DDDDDDDDDD+DDDDDDDD+++D++++++++++++++++++++++++++++=D
    5  D/++++++++++++++++DDDD+D+++++DDDDD+++++++++++++++++++++++++++DDDD+=D
    6  D/++++++++++++++++DD+D+++++++++++++++++++++++++++++++++++++DDDDDDDD
    7  DDDDDDDDDD++++++++++D++++++++++++++++++++++DDDDD+++++++DDDDDDDDD
    8  DDDDDDDDDDDDDDD++++++++++++++++++++++++++++DDDDDDDD+DDDDDDDDDDDDD
    9  DDDDDDDDDDDDDDD++++++++++DDD+++++++D+DDDDDDDDDDDDDDDDDDDDDDDDDDDDD
   10  DDDDDDDDDDDDDDD++++++++DDDDDDDDDDD+DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
   11  DDDDDDDDDDDDDDDD+++++++DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
   12  DDDDDDDDDDDDDDDDDDD++++DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
   13  DDDDDDDDDDDDDDDDDD+DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
   14  DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
```

HISTORY LOCATIONS

```
( 4, 2)    ( 5, 2)    ( 6, 2)    ( 5, 8)    ( 6,12)    ( 7,17)    ( 9,20)    (11,20)
( 7,28)    ( 7,36)    ( 5,44)    ( 5,52)    ( 3,60)
```

HISTORY LOCATIONS PLOTTED ON COMPUTATIONAL GRID

```
1, 2, ...  -  HISTORY LOCATIONS (ORDERED AS ABOVE, MOD 10)
        +  -  D(I,J) .GT. 0.
        D  -  D(I,J) .EQ. 0.
        -  -  D(I,J) .LT. 0.

      J          1         2         3         4         5         6
       123456789012345678901234567890123456789012345678901234567890123456789012345678901234
  I
   1   DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
   2   DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
   3   DDDDDD+D+++++DDDD+DDDDDDDDDDDDDDDDDDDDDDDD+++++D+DDDDDDDDD+++3++DD
   4   D1+++++++++++++D+DDDDDDDDDD+DDDDDDD+++D+++++++++++++D++++++++=D
   5   D2+++++4++++++++++DDDD+D+++++DDDDD++++++++1++++++2+++++DDDD+=D
   6   D3+++++++++5++++++++DD+D+++++++++++++++++++++++++++++++++DDDDDDDD
   7   DDDDDDDDDD++++++6+++D++++++9++++++++0++++++++DDDDD++++++++DDDDDDDDD
   8   DDDDDDDDDDDDDDD+++++++++++++++++++++++++++++DDDDDDDDD+DDDDDDDDDDDDD
   9   DDDDDDDDDDDDDDDD++++7+++++DDD++++++++D+D+DDDDDDDDDDDDDDDDDDDDDDDDDD
  10   DDDDDDDDDDDDDDDD+++++++++DDDDDDDDDD+DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
  11   DDDDDDDDDDDDDDDD+++8+++DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
  12   DDDDDDDDDDDDDDDDDD++++DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
  13   DDDDDDDDDDDDDDDDDDD+DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
  14   DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
```

DESIRED TIMES FOR SNAPSHOTS (TS(K),K=1,KS)

```
        11.000      40.000
```

-- NEAREST TIME STEPS

```
        16          58
```

-- ACTUAL TIMES FOR SNAPSHOTS

```
        11.005      39.892
```

----- FORCED LANDSLIDE -----

SLA91

VARIABLE	VALUE	DESCRIPTION
NSL	1	NUMBER OF SLICES IN LANDSLIDE
NXH	2	NUMBER OF HEIGHTS SPECIFIED FOR EACH SLICE
NTV	11	NUMBER OF VELOCITIES SPECIFIED FOR EACH SLICE
MSUB	-0	NUMBER OF SUBDIVISIONS PER GRID IN THE X-DIRECTION FOR LANDSLIDE CALCULATIONS
NSUB	-0	NUMBER OF SUBDIVISIONS PER GRID IN THE Y-DIRECTION FOR LANDSLIDE CALCULATIONS
THETA	0.	ANGLE FROM X-AXIS TO FORWARD DIRECTION OF SLIDE (DEGREES COUNTERCLOCKWISE)
WSL	1250.00	WIDTH OF SLIDE (DISTANCE UNITS)
XO	3.90000	X-COORDINATE OF RIGHT LEADING CORNER OF SLIDE (GRID UNITS, NOT DISTANCE)
YO	8.50000	Y-COORDINATE OF RIGHT LEADING CORNER OF SLIDE (GRID UNITS)
CDP	-0.	PRESSURE DRAG COEFFICIENT
CDV	-0.	VISCOUS DRAG COEFFICIENT
TSLCNV	15.8114	FACTOR TO CONVERT TSL(*) TO SECONDS
VSLCNV	13.8290	FACTOR TO CONVERT VSL(*) TO APPROPRIATE UNITS

DEFAULT VALUES SET FOR THE FOLLOWING INPUT VARIABLES

VARIABLE	VALUE
MSUB	50
NSUB	50

HEIGHT PROFILE (FUNCTION OF DISTANCE FROM LEADING EDGE)

I I=1,NXH	DISTANCE XSL(I)	HEIGHT (HSL(I,J),J=1,NSL) -- 1
1	0.	400.000
2	722.000	0.

VELOCITY PROFILE (FUNCTION OF TIME)

I I=1,NTV	TIME TSL(I)	VELOCITY (VSL(I,J),J=1,NSL) -- 1
1	0.	0.
2	.016	.240
3	.103	1.540
4	.190	2.480
5	.277	3.370
6	.364	3.790
7	.451	3.950
8	.538	3.660
9	.625	3.340
10	.712	2.720
11	.799	0.

VELOCITY PROFILE AFTER APPLYING CONVERSION FACTORS

I I=1,NTV	TIME TSL(I)	VELOCITY (VSL(I,J),J=1,NSL) -- 1
1	0.	0.
2	.253	3.319
3	1.629	21.297
4	3.004	34.296
5	4.380	46.604
6	5.755	52.412
7	7.131	54.624
8	8.507	50.614
9	9.882	46.189
10	11.258	37.615
11	12.633	0.

PLOT OF SLIDE PATH ON COMPUTATIONAL GRID

```
1, 2, ...  -  GRID ELEMENT CROSSED BY A SINGLE SLICE  (DIGIT IS SLICE NUMBER)
A, B, ...  -  GRID ELEMENT INITIALLY UNDER A SINGLE SLICE  (LETTERS CORRESPOND TO SLICE NUMBER, A=1, ETC.)
         /  -  GRID ELEMENT CROSSED BY, OR INITIALLY UNDER, TWO OR MORE SLICES
         +  -  COMPUTATIONAL ELEMENT NOT IN SLIDE PATH
   (BLANK)  -  DRY ELEMENT
```

```
  J          1         2         3         4         5         6
    123456789012345678901234567890123456789012345678901234567890123456789012345678901234
  I ----------------------------------------------------------------
   1-                                                                 -1
   2-                                                       ++++++    -2
   3-       + AAAAA    +                        +++++ +       ++++++   -3
   4- +++++++AAAAA+++ +          +        +++ +++++++++++++ +++++++++ -4
   5- +++++++11111++++++    + +++++     ++++++++++++++++++++++    ++ -5
   6- +++++++11111++++++  + ++++++++++++++++++++++++++++++++++        -6
   7-           ++++++++++ ++++++++++++++++++++++     +++++++         -7
   8-               ++++++++++++++++++++++++++++        +              -8
   9-                +++++++++   +++++++ + +                          -9
  10-                +++++++++          +                              -10
  11-                 +++++++                                         -11
  12-                  ++++                                           -12
  13-                   +                                             -13
  14-                                                                 -14
    ----------------------------------------------------------------
    123456789012345678901234567890123456789012345678901234567890123456789012345678901234
              1         2         3         4         5         6
```

TIME = 11.005 (SEC) SNAPSHOT OF SURFACE ELEVATION

I \ J	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DCD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD
2	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DCD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD
3	DDD	DDD	DDD	DDD	DDD	DDD	15.	DDD	34.	36.	37.	36.	34.	DDD	DCC	DDD	DDD	.00	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DCD	DDD	DDD	DDD
4	DDD	.00	.03	.37	3.1	16.	40.	45.	33.	50.	51.	46.	35.	46.	25.	4.8	DDD	.00	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	...	DDD
5	DDD	.01	.07	.68	5.2	27.	53.	-17.	35.	71.	67.	51.	34.	-9.9	47.	9.6	.58	.02	.00	DDD	DDD	DDD	DDD	...	DDD	...	...	...	...	...
6	DDD	.00	.02	.23	2.[illegible]	13.	58.	114	162	203	221	191	112	110	5[illegible].	7.7	.60	.03	.00	DDD	CDD	.00	DDD	...	...	...	...	...	...	...
7	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	186	166	111	66.	14.	1.7	.16	.01	.00	.00	DDD	.00	.00	.00	...	...	...	...	...	...
8	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	1.6	.14	.02	.00	.00	.00	.00	.00	.00	.00	.00	...	...	...	...	...
9	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DCD	.00	.00	.00	.00	.00	.00	.00	.00	.00	...	DDD	DDD	DDD	...	...
10	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	.00	.00	.00	.00	.00	.00	.00	.00	...	DDD	DDD	DDD	DDD	DDD	DDD
11	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DCD	DDD	.00	.00	.00	.00	.00	.00	...	DDD	DDD	DDD	DCD	DDD	DDD	DDD
12	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	.00	.00	.00	...	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD
13	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DCD	DDD	DDD	DDD	DDD	.00	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD
14	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD

I \ J	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
1	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD
2	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	...	...	...	...	...
3	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DCD	...	...	...	...	...	DDD	...	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	...	...	...	...
4	DDD	DDD	DDD	DDD	DDD	DDD	...	...	...	DDD	...	...	...	...	...	...	...	...	...	...	...	...	...	DDD	...	...	...	...	...	...
5	DDD	DDD	DDD	DDD	DDD	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	CCD	DDD	DDD
6	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	DDD	CCD	DDD	DCC
7	...	...	...	...	...	...	...	...	...	...	...	...	...	DDD	DDD	DDD	DDD	DDD	...	...	...	...	...	...	...	DDD	DDD	DDD	DDD	DDD
8	...	...	...	...	...	...	...	...	...	...	...	...	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	...	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD
9	...	...	...	...	...	DDD	...	DDD	...	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD
10	DDD	DDD	DDD	DDD	...	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDC	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	CDD	DDD	DDD
11	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD
12	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD
13	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	CDD	DDD	DDD
14	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DCC	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	DDD	CCD	DDD	DDC

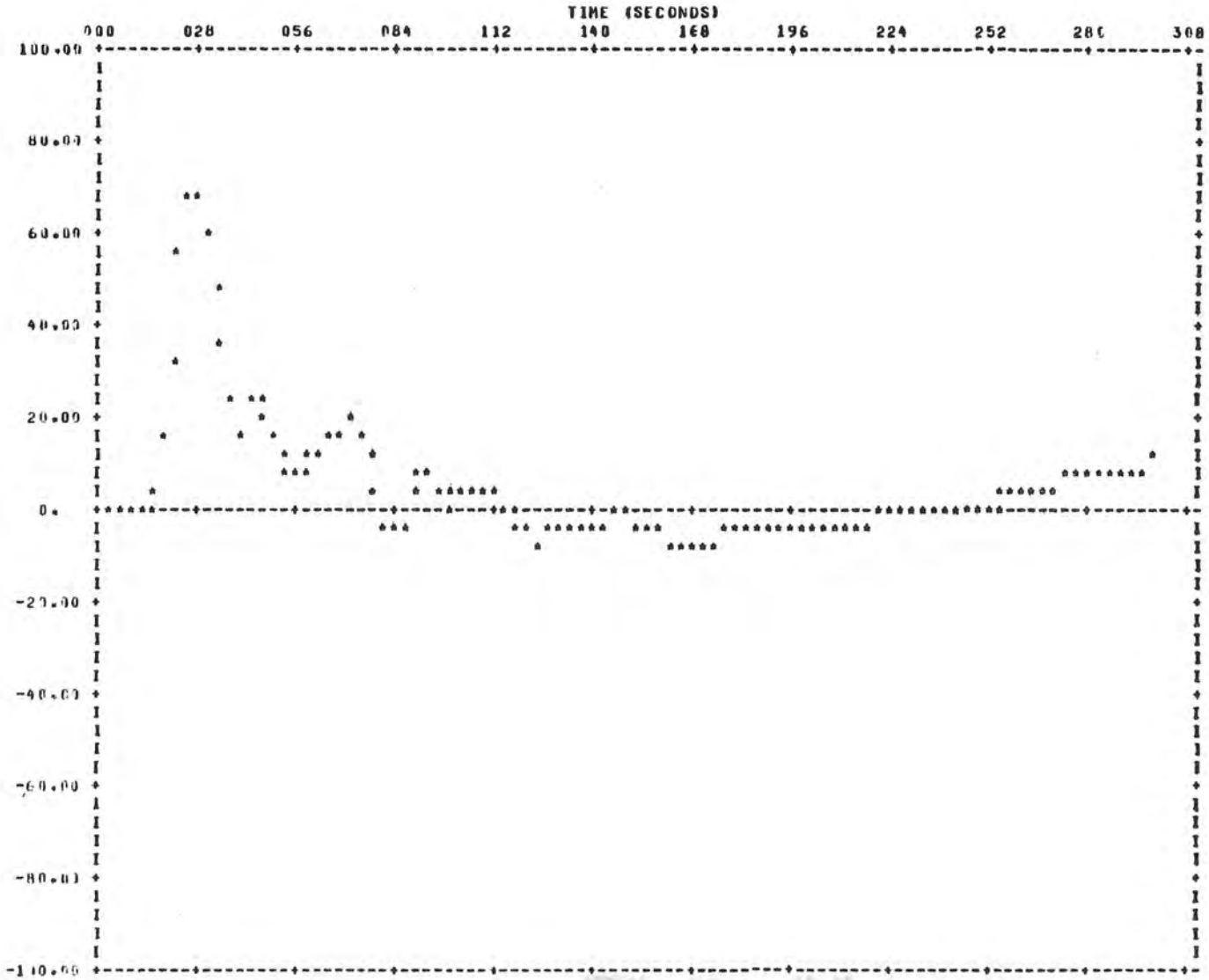
I \ J	61	62	63	64
1	DDD	DDD	DDD	DDD
2	...	DDD	DDD	DDD
3	...	...	DDD	DDD
4	...	...	...	DDD
5	DDD	...	...	DDD
6	DDD	DDD	DDD	DDD
7	DDD	DDD	DDD	DDD
8	DDD	DDD	DDD	DDD
9	DDD	DDD	DDD	DDD
10	DDD	DDD	DDD	DDD
11	DDD	DDD	DDD	DDD
12	DDD	DDD	DDD	DDD
13	DDD	DDD	DDD	DDD
14	DDD	DDD	DDD	DDD

... = 0. (EXACTLY)
/// OUT OF RANGE (ABS .GE. 999.5)
DDD DRY (NOT COMPUTED)

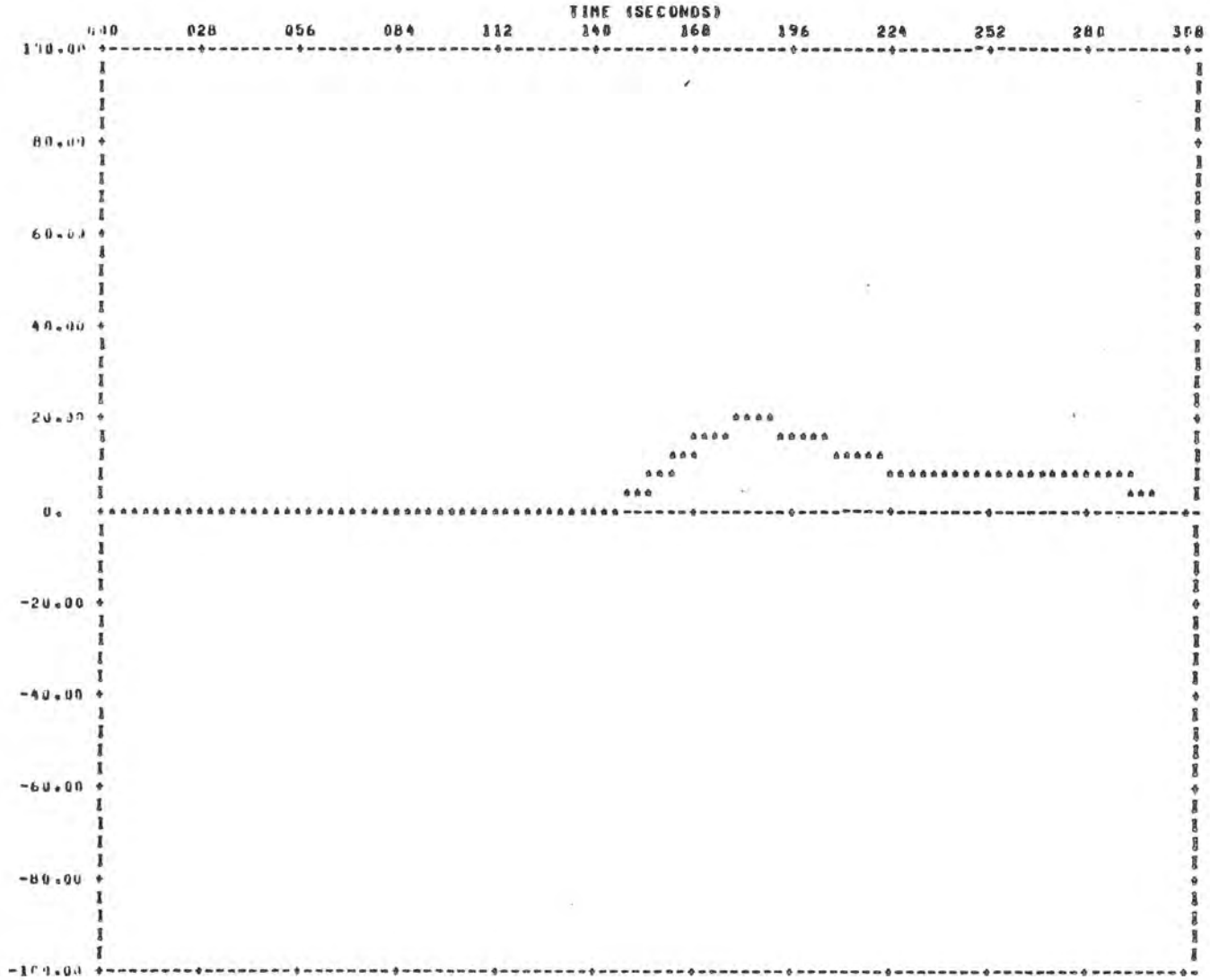
TIME STEP	TIME (SEC)	1 (4, 2) -E-	2 (5, 2) -E-	3 (6, 2) -E-	4 (5, 8) -F-	5 (6,12) -E-	6 (7,17) -E-	7 (9,20) -E-	8 (11,20) -E-	9 (7,28) -E-	10 (7,36) -E-
0	0.	0.	0.	0.	.016	.008	0.	0.	0.	0.	0.
3	2.163	0.	0.	0.	.897	.541	0.	0.	0.	0.	0.
6	4.127	.000	.000	0.	9.025	6.879	.000	0.	0.	0.	0.
9	6.190	.000	.000	.000	61.784	72.641	.001	0.	0.	0.	0.
12	8.254	.000	.001	.000	69.747	170.489	.024	.000	.000	0.	0.
15	10.317	.005	.015	.004	-37.524	166.653	.385	.000	.010	0.	0.
18	12.380	.008	.169	.061	22.110	12.245	4.491	.000	.000	.000	0.
21	14.444	.575	1.236	.560	62.099	-23.841	29.082	.001	.000	.000	0.
24	16.507	3.245	5.246	3.304	33.939	-11.126	63.539	.015	.000	.006	0.
27	18.570	11.851	14.631	12.340	-2.081	-3.911	20.351	.183	.003	.000	.000
30	20.634	26.337	31.145	29.854	-13.018	-9.053	8.771	1.783	.045	.000	.000
33	22.697	51.915	54.315	53.361	11.639	-10.073	54.574	10.711	.464	.000	.000
36	24.761	71.210	68.460	72.428	18.571	-11.062	28.142	34.972	3.529	.000	.000
39	26.824	74.544	69.444	72.552	3.701	-27.104	-5.017	37.627	17.464	.000	.000
42	28.887	67.993	68.032	64.356	17.468	-54.746	34.326	11.974	46.592	.000	.000
45	30.951	62.653	59.532	61.489	25.365	-74.262	32.096	16.111	27.672	.000	.000
48	33.014	51.236	46.977	51.322	7.122	-78.425	-10.392	38.999	3.145	.001	.000
51	35.078	35.018	35.852	35.850	4.136	-65.307	5.804	16.556	31.422	.009	.000
54	37.141	22.781	25.533	23.235	12.748	-39.772	28.139	5.638	55.612	.051	.000
57	39.204	16.531	16.595	16.409	23.155	-12.434	2.277	29.142	11.454	.242	.000
60	41.268	16.243	17.310	16.354	29.359	1.140	-15.651	21.977	8.045	.905	.000
63	43.331	21.651	23.866	22.549	28.431	-6.316	1.421	4.248	50.195	2.660	.000
66	45.394	25.401	24.186	26.195	31.469	-18.972	14.843	14.210	26.398	5.975	.000
69	47.458	21.511	19.324	21.271	32.499	-9.369	-2.215	21.683	.268	10.253	.000
72	49.521	14.981	14.868	14.214	23.766	34.509	-16.060	12.095	29.321	13.134	.000
75	51.585	10.931	11.186	10.593	21.058	82.654	-7.982	9.028	30.874	14.743	.001
78	53.648	8.812	9.130	9.037	26.354	9.498	-2.265	11.279	-4.664	15.446	.004
81	55.711	8.117	9.088	8.419	25.573	-16.764	-4.921	12.921	2.956	15.894	.017
84	57.775	8.971	9.393	9.008	20.367	3.115	1.471	10.945	23.318	17.692	.060
87	59.838	10.682	10.736	10.493	13.542	47.964	2.221	-.016	6.776	20.158	.180
90	61.902	12.957	13.109	12.741	5.260	89.162	-11.365	-6.647	-7.081	18.664	.472
93	63.965	14.737	14.366	14.564	-.344	17.202	-4.531	2.898	.644	16.167	1.001
96	66.028	15.339	15.239	15.227	-.526	-9.472	9.957	6.535	-4.909	18.925	2.167
99	68.092	16.473	17.012	16.527	4.127	-6.448	2.625	-5.225	-11.033	21.772	3.805
102	70.155	18.671	18.545	18.821	10.047	10.891	-.697	-10.851	1.965	17.305	5.918
105	72.218	19.636	18.603	19.645	11.868	27.888	6.593	-5.629	1.058	15.428	8.280
108	74.282	17.407	16.187	17.209	9.178	30.396	5.076	-.628	-20.096	18.362	10.663
111	76.345	11.873	10.669	11.589	7.094	19.463	10.626	-2.739	-18.279	18.253	12.624
114	78.409	4.468	3.639	4.224	7.456	1.415	12.968	-7.358	8.531	10.610	14.444
117	80.472	-2.788	-2.390	-2.883	7.863	-8.922	1.976	-4.290	4.959	14.753	15.279
120	82.535	-7.538	-5.991	-7.345	7.318	-7.283	6.161	4.778	-17.997	14.193	15.615
123	84.599	-7.956	-5.757	-7.410	7.111	-7.189	19.227	4.313	-13.575	4.616	16.160
126	86.662	-3.656	-2.288	-3.144	8.373	-1.644	15.408	-2.222	12.876	3.475	16.833
129	88.726	2.454	2.415	2.876	9.523	7.644	11.787	1.295	7.661	11.767	16.836
132	90.789	7.268	6.326	7.092	9.167	10.325	13.070	7.779	-12.451	8.285	16.585
135	92.852	8.133	6.769	7.569	7.839	2.821	11.518	7.144	3.807	-2.970	16.779
138	94.916	5.747	4.985	5.252	6.276	-3.986	5.316	7.463	20.366	-2.188	16.167
141	96.979	2.762	2.884	2.628	2.973	-.572	-1.835	8.497	1.213	5.793	14.472
144	99.042	1.401	1.995	1.576	-1.600	7.543	5.299	5.877	2.308	1.496	13.852
147	101.106	1.809	2.238	2.134	-4.963	9.905	14.996	7.008	25.689	-8.007	13.679
150	103.169	2.795	2.874	2.836	-5.892	7.564	-.646	7.254	12.979	-6.319	11.467
153	105.233	3.492	3.363	3.342	-4.715	7.204	-12.303	5.116	-1.303	.636	9.547
156	107.296	3.842	3.706	3.675	-2.323	8.561	1.305	6.648	14.550	-1.507	10.446
159	109.359	4.087	3.971	4.051	.421	8.178	13.545	7.851	16.981	-6.766	10.682
162	111.423	4.046	3.786	4.118	2.535	-.354	3.598	6.604	5.397	-5.237	6.179
165	113.486	3.149	2.749	3.207	2.770	-2.705	-6.432	7.180	8.629	-1.240	4.413
168	115.550	1.182	.655	1.171	1.051	-.945	-.298	6.741	10.252	-.169	6.642

TIME STEP	TIME (SEC)	I1 (5,44) -E-	I2 (6,52) -E-	I3 (3,60) -E-	Q *10**6	DISEL VOL *10**6
0	0.	0.	0.	0.	0.	19.803
3	2.763	0.	0.	0.	0.	32.364
6	4.127	0.	0.	0.	0.	78.368
9	6.190	0.	0.	0.	0.	138.533
12	8.254	0.	0.	0.	0.	176.747
15	10.317	0.	0.	0.	0.	197.542
18	12.380	0.	0.	0.	0.	197.835
21	14.444	0.	0.	0.	0.	0.
24	16.507	0.	0.	0.	0.	0.
27	18.570	0.	0.	0.	.078	0.
30	20.634	0.	0.	0.	.145	0.
33	22.697	0.	0.	0.	.585	0.
36	24.761	.000	0.	0.	1.290	0.
39	26.824	.000	0.	0.	1.694	0.
42	28.887	.000	0.	0.	1.568	0.
45	30.951	.000	.020	0.	1.379	0.
48	33.014	.000	.030	0.	1.123	0.
51	35.078	.000	.030	0.	.737	0.
54	37.141	.000	.030	.000	.397	0.
57	39.204	.000	.030	.000	.163	0.
60	41.268	.000	.000	.000	.098	0.
63	43.331	.000	.030	.000	.140	0.
66	45.394	.000	.030	.000	.239	0.
69	47.458	.000	.030	.000	.227	0.
72	49.521	.000	.030	.000	.144	0.
75	51.585	.000	.030	.000	.063	0.
78	53.648	.000	.030	.000	.028	0.
81	55.711	.000	.030	.000	.016	0.
84	57.775	.000	.030	.000	.015	0.
87	59.838	.000	.000	.000	.022	0.
90	61.902	.000	.000	.000	.041	0.
93	63.965	.000	.000	.000	.066	0.
96	66.028	.000	.000	.000	.082	0.
99	68.092	.000	.030	.000	.094	0.
102	70.155	.000	.030	.000	.119	0.
105	72.218	.000	.030	.000	.146	0.
108	74.282	.000	.000	.000	.141	0.
111	76.345	.000	.000	.000	.089	0.
114	78.409	.000	.030	.000	.021	0.
117	80.472	.000	.000	.000	0.	0.
120	82.535	.000	.000	.000	0.	0.
123	84.599	.212	.000	.000	0.	0.
126	86.662	.445	.030	.000	0.	0.
129	88.726	.048	.030	.000	0.	0.
132	90.789	1.493	.030	.000	.000	0.
135	92.852	2.427	.000	.000	.016	0.
138	94.916	3.665	.000	.000	.004	0.
141	96.979	5.147	.000	.000	0.	0.
144	99.042	6.757	.000	.000	0.	0.
147	101.106	8.351	.000	.000	0.	0.
150	103.169	9.451	.022	.000	0.	0.
153	105.233	11.247	.049	.000	0.	0.
156	107.296	12.567	.103	.000	0.	0.
159	109.359	13.652	.214	.000	0.	0.
162	111.423	15.115	.378	.000	0.	0.
165	113.486	16.264	.662	.000	0.	0.

SURFACE ELEVATION LOCATION 2 GRID (5, 2)

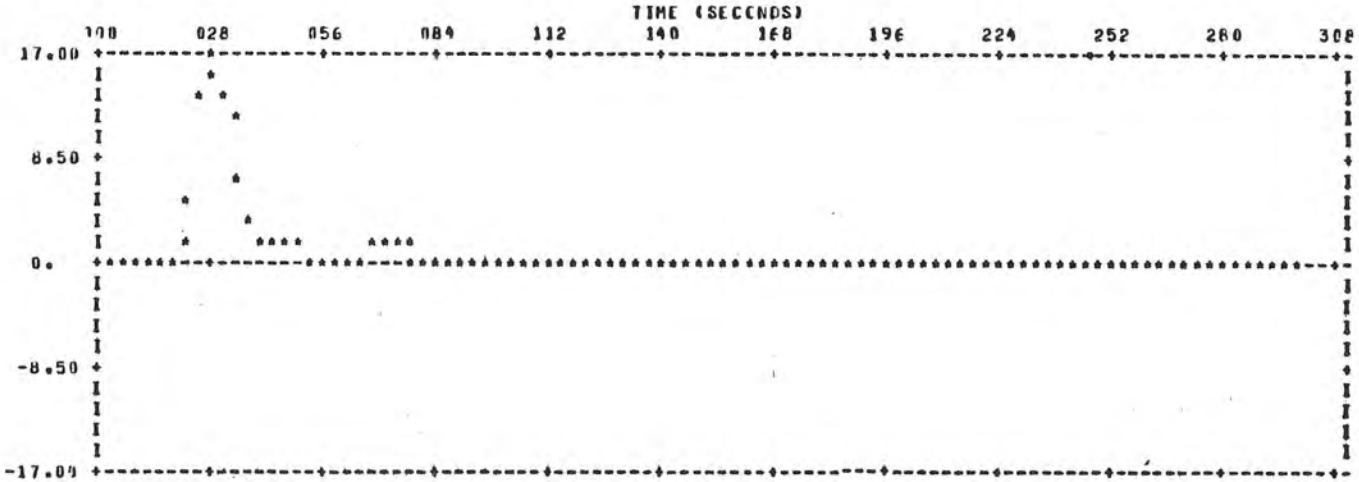


SURFACE ELEVATION LOCATION 13 GRID 6 3.6E0



OVER-TOPPING RATE

MULTIPLIER = 1.0E+05



TOTAL VOLUME OVER-TOPPING = 2.29033893E+07

APPENDIX B1. SUMMARY OF MODIFICATIONS TO DAMBRK

The time history of the discharge over a dam, as computed in the numerical model LSWAVE, is stored on a disk or magnetic tape file. This data file can be fed into a downstream flood routing model to simulate the propagation of the flood due to the landslide-induced overtopping.

A flood routing model developed in the National Weather Service, called DAMBRK, was adopted in this study to illustrate the procedure of transferring the discharge hydrograph from LSWAVE to a downstream model. The version B of DAMBRK, which is a short version of version A, dated September 10, 1980, was used. Before the original program was stored into the program library, a program statement was added to the beginning of the main program, and a "*DECK" card was added to the beginning of the main program and every subroutine.

After the whole program was stored in the program library, it was modified through the UPDATE command available in the CDC 6600/7600 system, such that the modified program could be used to read the file generated from LSWAVE and simulate the propagation of short-period huge waves as happened in this study. A listing of the control cards for "UPDATE" to modify the original program is provided in the following two pages. In this list, the number used in the INSERT and DELETE commands are the sequential number of the program statements in the associated subprogram. The sequential numbers start with 2 for the first statement of each subroutine. In the main program, the sequential numbers start with 2 for the program statement added in this study and so the original first statement has a sequential number of 3.

```
*IDENT SUPMOD
*DELETE DAMBRK.2
      PROGRAM DMBRKB(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1)
*INSERT DAMBRK.6
C                                MODIFIED BY TETRA TECH, INC., NOV. 15, 1981
*DELETE DAMBRK.11
      COMMON /SIM/ Q(700),TI(503),GI(503),YO,RLM,SCM,CMA,YBMIN,BB,Z,TFH,
*DELETE      DAMBRK.30,DAMBRK.40
      COMMON XS(90),X(200),NT(6),NN(90),BST(8,90),BSST(8,90),
     A      HST(8,90),FS(700),YI(200),DXM(90),CMT(8,200),FFKC(90),
     B      DSA(8,2),SSA(8,2),YM(200),QMX(200),TM(200),YMG(200),GS(700),
     C      MESAGE(20),FSTG(90),NTFS(6),YDT(90),TTPT(700),VERC(200),
     D      QCM(700,6),YUU(200),QUU(200),TFLOD(200),FLOCH(200),
     E      FLODHT(90),NPT(30),ICT(10),GOT(10),MS(6)
*DELETE DAMBRK.192
      READ(LU5,10)MDAM,MRVR,MNAME
      IF (EOF(LU5)) 1350,6,1350
    6 CONTINUE
*DELETE DAMBRK.202
      READ(LU5,10) MDAM,MRVR,MNAME
      IF (EOF(LU5)) 1350,11,1350
*INSERT DAMBRK.214
      KKNS=KKN
      KKN=IABS(KKN)
*DELETE DAMBRK.475
     1F8.5,2X,4HHRS,//9X,44HTEH(TIME AT WHICH COMPUTATIONS TERMINATE) =
*DELETE DAMBRK.484,DAMBRK.485
      IF (KKNS .GT. 0) GOTO 165
      READ (5,1300) QIMIN
      READ (1) TZERO,DTIS,ITEH,(QI(I),I=1,ITEH)
      IF(ITEH.LE.500) GO TO 138
      WRITE(6,139)
  139 FORMAT(///# ITEH.GT.500... CHANGE DIMENSIONS OF GI AND TI IN COMMO
     1N/SIM/,#/# ALSO CHANGE THE NUMBER 500 IN THE STATEMENT BEFORE STAT
     2EMENT NUMBER 139.#)
      STOP
  138 DTIS=DTIS/3600.
      IF (TZERO .EQ. 0.) GOTO 150
      K9=ITEH
      ITEH=ITEH+1
      JACK=ITEH
      DO 140 K=1,K9
      GI(JACK)=AMAX1(QI(JACK-1),QIMIN)
  140 JACK=JACK-1
      GI(1)=QIMIN
      TI(1)=0.
      TI(2)=TZERO/3600.
      DO 145 K=3,ITEH
  145 TI(K)=TI(K-1)+DTIS
      GOTO 160
  150 TI(1)=0.
      QI(1)=AMAX1(GI(1),QIMIN)
      DO 155 K=2,ITEH
      QI(K)=AMAX1(QI(K),QIMIN)
  155 TI(K)=TI(K-1)+DTIS
  160 IF (DTIS*ITEH .GE. TEH) GOTO 170
      ITEH=ITEH+1
      GI(ITEH)=QIMIN
      TI(ITEH)=TI(ITEH)+DTIS
      IF (DTIS*ITEH .GE. TEH) GOTO 170
      ITEH=ITEH+1
      GI(ITEH)=QIMIN
      TI(ITEH)=TEH
      GOTO 170
  165 READ (5,1300) (QI(K),K=1,ITEH)
      IF(DHF.LT.0.0001) READ(5,1300) (TI(K),K=1,ITEH)
  170 NU=ITEH
```

```
*DELETE DAMBRK.496
      IF(DHF.GT.0.0001) GO TO 183
*DELETE DAMBRK.500
 182  FORMAT(12X,8F8.5)
*DELETE DAMBRK.521
  190 READ(LU5,1290) NS,NCS,NTT,JNK,KSA,KSUPC,LQ
*DELETE DAMBRK.781
      IF(DTHM.GT.0.00001) JDTH=1
*DELETE DAMBRK.1497
      ITMAX=20
*INSERT DAMBRK.1524
      TFH=0.
      HD=0.
*DELETE DAMBRK.1536
      IF(TFH.LT.0.0001) TFH=10.
*DELETE DAMBRK.1587
     4   10X,15HTIME TO FAILURE,31X,2HHR,6X,3HTFH,4X,F10.6//
*DELETE DAMBRK.1590
     7   10X,12HTIME TO PEAK,34X,2HHR,6X,2HTP,5X,F10.6//
*DELETE DAMBRK.1591
     8   10X,14HTIME STEP SIZE,32X,2HHR,6X,4HDTHI,3X,F10.6////)
*DELETE DAMBRK.1756
      CALL INTER(J,F1,IDAM)
*DELETE DAMBRK.1847
 1070 FORMAT(10X,5HTT= ,F8.6,11X,6HDTH = ,F9.6,11X,8HITTER = ,I3/
*DELETE DAMBRK.1997
 1165 FORMAT(//10X,6HKTIME=,I3,5X,16HALLOWABLE KTIME=,I4,10X,3HTT=,F8.6)
*DELETE DAMBRK.2115
 1265 FORMAT(/////5X,5HTP = ,F10.8,10X,7HTFOO = ,F10.8,10X,7HQMAX = ,
*DELETE DAMBRK.2116
     1F12.2,10X,6HTFH = ,F10.8/////)
*INSERT DAMBRK.2119
      IF(KKN.EQ.9) GO TO 1350
*DELETE DAMBRK.2125
 1280 IF(KKN.EQ.9) GO TO 1350
*DELETE DAMBRKC.4
      COMMON /SIM/ Q(700),TI(503),GI(503),YO,RLM,SOM,CMN,YBMIN,BB,Z,TFH,
*DELETE BDARY.4
      COMMON /SIM/ Q(700),TI(503),GI(503),YO,RLM,SOM,CMN,YBMIN,BB,Z,TFH,
*DELETE INTER.4
      COMMON /SIM/ Q(700),TI(503),GI(503),YO,RLM,SOM,CMN,YBMIN,BB,Z,TFH,
*DELETE INTERP.3
      COMMON /SIM/ Q(700),TI(503),GI(503),YO,RLM,SOM,CMN,YBMIN,BB,Z,TFH,
*INSERT PLOT.82
      IF(HDIF.LT..1) HINCR=.005
      IF(HDIF.LT..05) HINCR=.001
      IF(HDIF.LT..025) HINCR=.0005
*INSERT PLOT.192
      IF(HINCR.EQ..0005) WRITE(6,118) H,S,IQ,MPLOT
      IF(HINCR.EQ. .001) WRITE(6,117) H,S,IQ,MPLOT
      IF(HINCR.EQ. .005) WRITE(6,117) H,S,IQ,MPLOT
  117 FORMAT(2X,F7.3,F7.1,I9,2X,51A1)
  118 FORMAT(1X,F8.4,F7.1,I9,2X,51A1)
*INSERT PRPLOT.13
      DATA PK/C./
```

LIST OF REFERENCES

Balloffet, A.; Cole, E.; & Balloffet, A., 1974, "Dam collapse wave in a river," J. Hyd. Div., ASCE, 100 (HY5), Proc. Paper 10523, 645-665.

Batchelor, G.K., 1967, An Introduction to Fluid Dynamics, xviii+615pp, Cambridge Univ. Press, Cambridge, U.K.

Bradley, J.N., 1952, "Discharge coefficients for irregular overfall spillways." U.S. Bur. Reclamation, Eng. Monographs, No. 9.

Brandsma, M.; Divoky, D.; & Hwang, L.-S., 1974, "Response of small islands to long waves," Tetra Tech Report No. TC-349, Tetra Tech, Inc., Pasadena, Calif.

Brater, E.F. & King, H.W., 1976, Handbook of Hydraulics for the Solution of Hydraulic Engineering Problems, 6th ed., McGraw Hill, N.Y., N.Y.

Chen, C. & Armbruster, J.T., 1980, "Dam-break wave model: Formulation and verification," J. Hyd. Div., ASCE, 106 (HY5), Proc. Paper 15395, 747-767.

Chiang, W.-L., 1977, "Further development and testing of a stream-acquifer system model," Master's thesis, Univ. of Kansas, Lawrence, Ks.

Chiang, W.-L., 1979, "Tide-Induced Currents in Harbors of Arbitrary Shape," Ph.D. dissertation, Univ. of Southern Calif., Los Angeles, Calif., ix+194pp.

Chow, V.T., 1959, Open-Channel Hydraulics, xviii+680pp., McGraw Hill, N.Y., N.Y.

Chow, V.T. (ed.), 1964, Handbook of Applied Hydrology, A Compendium of Water-resources Technology, McGraw Hill, N.Y., N.Y.

Daily, J.W. & Harleman, D.R.F., 1966, Fluid Dynamics, ix+454pp, Addison Wesley, Reading, Mass.

Das, M.M. & Wiegel, R.L., 1971, "Water waves generated by horizontal motion of a vertical wall," HEL-16-7, Univ. of Calif., Berkeley, Calif.

Davidson, D.D. & McCartney, B.L., 1975, "Water waves generated by landslides in reservoirs," J. Hyd. Div., ASCE, 101(HY12), 1489-1501.

LIST OF REFERENCES (continued)

Davidson, D.D. & Whalin, R.W., 1974, "Potential landslide-generated water waves, Libby Dam Lake Koocanusa, Montana; Hydraulic model investigation," TR-H-74-15, Army Engineer Waterways Experiment Station, Vicksburg, Miss.

Fread, D.L., 1980, "Two Finite-difference hydrodynamic models used for river flood forecasting by the National Weather Service," Lecture notes presented at Short Course on Computational Hydraulics, Texas A&M Univ., College Station, Texas, May 26-30, 1980.

Fread, D.L. & Harbaugh, T.E., 1973, "Transient hydraulic simulation of breached earth dam," J. Hyd. Div., ASCE, 99(HY1), Proc. Paper 9466, 139-154.

Fread, D.L. & Smith, G.F., 1978, "Calibration technique for 1-D unsteady flow models," J. Hyd. Div., ASCE, 104 (HY7), Proc. Paper 13892, 1027-1044.

Gilcrest, B.R., 1950, "Flood routing," In: Rouse, H. (ed.), Engineering Hydraulics, John Wiley & Sons, N.Y., N.Y.

Granville, P.S., 1979, "Drag", in Grady, R.J. (ed), "Hydroballistics design handbook, Volume 1," SEAHAC TR 79-1, Ch. 3, 309-341, Naval Sea Systems Command.

Henderson, F.M., 1966, Open Channel Flow, The Macmillan Co., N.Y., N.Y.

Hoerner, S.F., 1965, Fluid-dynamic Drag, Hoerner Fluid Dynamics.

Horton, R.E., 1907, "Weir experiments, coefficients, and formulas." U.S.G.S. Water-Supply Paper 200.

Hwang, L.-S. & Divoky, D., 1970, "Tsunami generation," J. Geophys. Res., 75(33), 6802-6817.

Hwang, L.-S.; Divoky, D.; & Butler, L., 1970, "Numerical studies of surface wave generation by bottom motion," Draft Final Report, TETRAT-P-71-174-3, x+52pp, Tetra Tech Inc., Pasadena, Calif.

Jolly, J.P. & Yevjevich, V., 1974, "Simulation accuracies of gradually varied flow," J. Hyd. Div., ASCE, 100 (HY7), Proc. Paper 10677, 1011-1030.

Jørstad, F.A., 1968, "Waves generated by landslide in Norwegian fjords and lakes," Oslo Norges Geotekniske Institutt, Publ. 79, 13-32.

Kamphius, J.W. & Bowering, R.J., 1970, "Impulse waves generated by landslides," Proc. 12th Coastal Engr. Conf., Sept. 13-18, 1970, Washington, D.C., 1, Ch. 35, 575-588.

Kandaswamy, P.K. & Rouse, H., 1957, "Characteristics of flow over terminal weirs and sills," J. Hyd. Div., ASCE, 83(HY4), Paper 1345, 1345-1-1345-12.

Kennard, E.H., 1949, "Generation of surface waves by a moving partition," Quar. Appl. Math., 7(3), 303-312.

Kiersch, G.A., 1964, "Vaiont reservoir desaster," Civil Engineering, 34(3), 32-39.

Kirkpatrick, K.W., 1957, "Discharge Coefficients for spillways at TVA dams," Transactions, ASCE, 122, Paper No. 2855, 190-210.

Koutitas, C.G., 1977, "Finite element approach to waves due to landslides," J. Hyd. Div., ASCE, 103(HY9), Proc. Paper 13218, 1021-1029.

Kranzer, H.C. & Keller, J.B., 1958, "Water waves produced by explosions," J. Appl. Phy., 30(3), 398-407.

Kuba, E.M., 1968, "A hydraulic model study of landslide-generated water waves in bays," HEL-19-2, Univ. of Calif., Berkeley, Calif.

Linsley, R.K. Jr.; Kohler, M.A.; & Paulhus, J.L.H., 1958, Hydrology for Engineers, McGraw-Hill, N.Y., N.Y.

McCulloch, D.S., 1966, "Slide-induced waves, seiching and ground fracturing caused by the earthquake of March 27, 1964 at Kenai Lake, Alaska," U.S. Geological Survey Professional Paper 543-A, A1-A41.

Madsen, O.S., 1970, "Waves generated by a piston-type wavemaker," Proc. 12th Coastal Engr. Conf., Sept. 13-18, 1970, Washington, D.C., 1, Ch. 36, 589-607.

Mercer, A.G.; Chaudhry, M.H.; & Cass, E.D., 1979, "Modeling of slide-generated waves," Proc. 4th Hydrotechnical Conf., May 1979, Vancouver, Cananda, 2, 730-745.

Merritt, F.S. (ed.), 1976, Standard Handbook for Civil Engineers, 2nd ed., McGraw Hill, N.Y., N.Y.

Miller, D.J., 1960, "Giant waves in Lituya Bay, Alaska," U.S. G.S. Professional Paper 354-C.

Muller, L., 1964, "The rockslide in the Vaiont Valley," Rock Mech. Eng. Geol., 2, 148-212.

Noda, E., 1969, "Theory of water waves generated by a time-dependent boundary displacement," Technical Report HEL 16-5, Univ. of Calif., Berkeley, Calif.

Noda, E., 1970, "Water waves generated by landslides," J. Waterways, Harbors and Coastal Engr. Div., ASCE, 96(WW4), Proc. Paper 7699, 835-855.

Northwest Hydraulic Consultants Ltd., 1976, "Final report, Hydraulic model study of waves from Downie Slide," Northwest Hydraulic Consultants Ltd., Edmonton, Vancouver, Canada.

Pogge, E.C., 1968, "Effects of bank seapage on flood hydrograph," Contribution No. 33, Kansas Water Resources Res. Inst., Lawrence, Ks.

Pugh, C.A. & Harris, D.W., 1982, "Prediction of landslide-generated water waves," Paper submitted to 14th ICOLD Congress on Large Dams, Brazil, 1982.

Rajar, R., 1978, "Mathematical simulation of dam-break flow," J. Hyd. Div., ASCE, 104(HY7), Proc. Paper 13883, 1011-1026.

Raney, D. & Butler, H.L., 1975, "A numerical model for predicting the effects of landslide-generated water waves," Waterways Experiment Station, Report H-75-1.

Rehbock, T., 1929, Discussion of Schoder, E.W. & Turner, K.B., "Precise weir measurements," Transactions, ASCE, 93.

Rosenhead, L. (ed.), 1963, Laminar Boundary Layers, xxxi+687pp, Oxford Univ. Press, Oxford, U.K.

Safety of Dams Evaluation Team for Morrow Point Dam, 1979, "Report on safety of dam and appurtenant works, Morrow Point Dam, Curecanti Unit, Colorado River Storage Project, Upper Colorado Region," U.S. Dept. of the Interior, Bur. Reclamation, Engr. and Res. Center, Denver, Colo.

LIST OF REFERENCES (continued)

Sakkas, J.G. & Strelkoff, T., 1973, "Dam-break flood in a prismatic dry channel," J. Hyd. Div., ASCE, 99(HY12), Proc. Paper 10233, 2195-2216.

Schlichting, H., 1968, Boundary Layer Theory, trans. from Germany by Kestin, J., 6th ed., xix+747pp, McGraw Hill, N.Y., N.Y.

Shreve, R., 1968, "Leakage and fluidization in air-layer lubricated avalanches," Bulletin, Geol. Soc. America, 79.

Slingerland, R.L. & Boight, B., 1979, "Occurrences, properties, and predictive models of landslide-generated water waves," In: Voight, B. (ed.), Rockslides and Avalanches, 2, Ch. 9, 317-397, Elsevier, Amsterdam, Netherlands.

Stelson, T.E. & Mavis, F.T., 1957, "Virtual mass and acceleration in fluids," Transacations, ASCE, 122, Paper No. 2879, 518-525.

Stoker, J.J., 1957, Water Waves, Interscience, N.Y., N.Y.

Strelkoff, T., 1969, "One-dimensional equations of open-channel flow," J. Hyd. Div., ASCE, 95(HY3), Proc. Paper 6557, 861-876.

Su, S.-T. & Barnes, A.H., 1970, "Geometric and frictional effects on sudden releases," J. Hyd. Div., ASCE, 96 (HY11), Proc. Paper 7650, 2185-2200.

Tetra Tech, Inc., 1975, "Estimation of water wave heights at Baker Lake resulting from possible landslides from Mt. Baker," Report No. TC-628, Tetra Tech, Inc., Pasadena, Calif.

U.S. Army Coastal Engineering Research Center, 1977, Shore Protection Manual, Vol. II, 3rd ed.

U.S. Department of the Interior, 1948, "Studies of crests for overfall dams," Boulder Canyon Project Final Report, Part VI, Hydraulic Investigations, Bulletin 3, Bur. Reclamation, Denver, Colo.

U.S. Department of the Interior, 1973, Design of Small Dams, 2nd ed., U.S. Government Printing Office, Washington, D.C.

Vasilier, O.F., 1971, "Numerical solution of the non-linear problems of unsteady flows in open channels," In: Holt, M. (ed.), Proc., 2nd Int. Conf. on Numerical Methods in Fluid Dynamics, Sept. 15-19, 1970, Univ. of Calif., Berkeley, Springer-Verlag.

Weinmann, P.E. & Laurenson, E.M., 1979, "Approximate flood routing methods: A review," J. Hyd. Div., ASCE, 105 (HY12), Proc. Paper 15057, 1521-1536.

Western Canada Hydraulic Laboratories, 1970, "Hydraulic model studies, Wave action generated by slides into Mica Reservoir," Report, Western Canada Hydraulic Laboratories, Vancouver, B.C., Canada.

Wiegel, R.L., 1955, "Laboratory studies of gravity waves generated by the movement of a submerged body." Trans. Am. Geophy. Union, 36(5), 759-774.

Wiegel, R.L.; Noda, E.K.; Kuba, E.M.; Gee, D.M. & Tornberg, G.F., 1970, "Water waves generated by landslides in reservoirs," J. Waterways, Harbors and Coastal Engr. Div., ASCE, 96(WW2), Proc. Paper 7279, 307-333.

Wu, T.H., 1966, Soil Mechanics, viii+431pp, Allyn and Bacon, Boston, Mass.

Xanthopoulos, Th. & Koutitas, Ch., 1976, "Numerical simulation of a two dimensional flood wave propagation due to dam failure," J. Hyd. Res., 14(4), 321-331.

Yen, B.C., 1973, "Open-channel flow equations revised," J. Engr. Mech. Div., ASCE, 99(EM5), Proc. Paper 10073, 979-1009.

Zitta, V.L. & Wiggert, J.M., 1971, "Flood routing in channels with bank storage," Water Resources Res., 7(5), 1341-1345.