SR 77-10

Special Report 77-10

# A COMPUTER ROUTING OF UNSATURATED FLOW THROUGH SNOW

Walter B. Tucker III and Samuel C. Colbeck

May 1977

Prepared for
DIRECTORATE OF FACILITIES ENGINEERING
OFFICE, CHIEF OF ENGINEERS
By
CORPS OF ENGINEERS, U.S. ARMY
COLD REGIONS RESEARCH AND ENGINEERING LABORATORY
HANOVER, NEW HAMPSHIRE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER<br>Special Report 77-10 | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle)<br>A COMPUTER ROUTING OF UNSATURATED FLOW THROUGH SNOW | | 5. TYPE OF REPORT & PERIOD COVERED |
| | | 6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s)<br>Walter B. Tucker III and Samuel C. Colbeck | | 8. CONTRACT OR GRANT NUMBER(s) |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>U.S. Army Cold Regions Research and Engineering Laboratory<br>Hanover, New Hampshire 03755 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS<br>DA Project 4A161102AT24<br>Task A1, Work Unit 001 |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Directorate of Facilities Engineering<br>Office, Chief of Engineers<br>Washington, D.C. 20314 | | 12. REPORT DATE<br>May 1977 |
| | | 13. NUMBER OF PAGES<br>43 |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) | | 15. SECURITY CLASS. (of this report)<br>Unclassified |
| | | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer programs
Forecasting
Snow
Snow runoff
Waterflow

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Computer programs for routing the vertical movement of water through snow have been developed. Previously, manual graphical techniques were used, making calculation of the flow and shock front a very time-consuming procedure. The shock front is dependent on surface melt taking place now as well as the antecedent flow in the snow, usually a function of the nature of the flow for the previous day. One program, designed to accommodate actual surface melt data, has the ability to handle complicated input profiles such as when melt is

DD FORM 1473, 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract (cont'd)

erratic on a cloudy day, creating such complexities as intersecting shock fronts. Another program, designed for rapid simulation purposes, approximates a simple surface input with a function, in this case a sine wave. This function is easily changed, allowing a variety of conditions to be assessed, although only one shock front is accommodated. Error analysis and some applications of the programs are presented.

## Preface

This report was prepared by Walter B. Tucker, III, Geologist, and Dr. Samuel C. Colbeck, Jr., Geophysicist, Snow and Ice Branch, Research Division, U.S. Army Cold Regions Research and Engineering Laboratory.

The work covered by this report was funded under DA Project 4A161102AT24, Research in Snow, Ice and Frozen Ground; task A1, Properties of Cold Regions Materials; work unit 001, Properties of Snow and Ice.

TABLE OF CONTENTS

# A COMPUTER ROUTING OF UNSATURATED FLOW THROUGH SNOW

by

Walter B. Tucker III

and

Samuel C. Colbeck

## Introduction

The need to make accurate forecasts of runoff from snowcovers has necessitated extensive investigations of the properties of seasonal snowcovers (e.g., Corps of Engineers 1956). Much information about the mode of flow of water through snow was generated by research studies starting 30 years ago and this information has been used in the formulation of hydrological forecasting models (e.g., U.S. Army Engineer Division 1972). Anderson's research model (Anderson 1973) uses a specific lag-concentration relationship which was obtained from site-specific studies. This relationship provides an empirical basis for routing the flow but cannot be readily generalized to include the properties of the snow. For example, the effects of layering, depth, density and grain size should be included implicitly in a forecasting scheme because these parameters are highly variable over the lifetime of a seasonal snowcover.

A physical basis for understanding the movement of water through snow has been developed (e.g., Colbeck, in press). The more-or-less vertical movement of water through snow can be described as unsaturated flow through porous media. This flow u is described by

$$3\,\alpha^{1/3}\,k^{1/3}\,u^{2/3}\,\frac{\partial u}{\partial z} + \phi_e\,\frac{\partial u}{\partial t} = 0 \qquad (1)$$

which has the solution

$$\left.\frac{dz}{dt}\right|_u = 3\alpha^{1/3}\,\frac{k^{1/3}}{\phi_e}\;u^{2/3} \qquad (2)$$

where $\left.\frac{dz}{dt}\right|_u$ is the downward movement of a value of u, $k^{1/3}/\phi_e$ represents the properties of the snow, and $\alpha$ is a constant. This solution can be applied directly to the decreasing surface input following the peak melting rate to explain why smaller flow rates travel more slowly, thus taking longer to reach the bottom of the snowcover. The difficulty with applying eq. 2 is that, during periods when the surface melting is increasing with time, slower moving (smaller) values of flux u are overtaken by faster moving (larger) values.

As shown in Figure 1, the intersecting values of flux join to form a shock front whose slope $d\xi/dt$ is given by

$$\frac{d\xi}{dt} = \alpha^{1/3}\,\frac{k^{1/3}}{\phi_e}\,(u_+^{2/3} + u_+^{1/3}\,u_-^{1/3} + u_-^{2/3}) \qquad (3)$$

where $u_+$ and $u_-$ are the larger and smaller values of flux which form the shock. The construction of the shock front is a slow exercise because of the need to use small intervals which minimize the interpolation errors. Even for research purposes, the construction of the diagrams by hand is very limiting and, for the purposes of hydrological forecasting, it is necessary to accomplish quickly this single part of the long routing procedure. Accordingly, a computer simulation of this water routing is developed here.

The computer program has the ability to handle a variety of situations, including complicated surface inputs such as multipeaked inputs to simulate melting on a partly cloudy day. Because the program is designed to handle most conceivable situations, the complete program is lengthy. For research purposes, many problems can be handled without using the entire program. A guide to the different aspects of the program and some information about the optimum step size for economical use of the program are given later.

## Description of Techniques

*Graphical Construction*. Given the parameter which characterizes the properties of the snow $k^{1/3}/\phi_e$ and the surface melting as a function of time, only an initial condition for flow is needed to construct the characteristics and shock front. This initial condition is the antecedent flow in the snow at the time the construction is started, usually the time at which the surface flux begins. The antecedent flow is generally determined by the nature of the flow during the previous day. Usually the antecedent flow increases with depth, although if no input has occurred at the surface for some time, the antecedent flow may be essentially zero.

Given the boundary and initial conditions, values of flux u can be attached to the t and z axes respectively (see Fig. 1). The values along the t axis represent the boundary condition u(o,t) and the values along the z axis represent the initial condition u(z,o). From the points on these axes which represent specific values of flux, the characteristic lines are constructed using eq. 2 to determine the slope of the line for each value of u. The values of u chosen for this construction are arbitrary, but the increments must be sufficiently small to allow an accurate interpolation between the characteristics lines.

When flux is an increasing function of time, the characteristics intersect as shown in Figure 1 where the characteristics from the initial condition intersect the characteristics from the boundary condition. The shock front in Figure 1 begins at the surface at the onset of surface melting because the characteristics intersect immediately upon the onset of surface melting. The intersecting characteristics determine the slope of the shock front at each point according to eq. 3. Once the shock front begins, it is constructed iteratively using the smallest increments practical and using a great deal of judgment to interpolate between the characteristic lines. While the computer can quickly handle many calculations with small increments, it is difficult to program the computer to have good judgment.

Once the shock front and characteristics are constructed for the z-t space of interest, the flow as a function of time at any depth (or the flow as a function of depth for any time) can be taken immediately from the z-t field. This is done by simply reading the values of flux which cross the depth (or time) line of interest. The time (or depth) at which the shock arrives can also be read immediately from the graph; but the strength of the shock, i.e., the maximum and minimum values of flux which define the shock, requires some interpolation between two characteristic lines on either side of the shock.

Computer Technique. Computer programs for finding runoff at depth for two general cases have been prepared. The first program was designed to accommodate actual digitized surface runoff data with the ability to handle multiple peaked surface inputs, intersecting shock fronts and the like. The other program is intended to simulate or approximate simple surface input and can only handle one shock front. Accordingly, the input must be characterized by some relatively simple function of time (e.g., sine wave). This program is somewhat faster and more accurate than the first and does not require extensive preparation of input data prior to execution.

In either case, the surface runoff as a function of time (boundary condition), the antecedent flow taking place when the calculation begins (initial condition), and the parameter $k^{1/3}/\phi_e$ which governs snow properties are required to calculate the flow at depth. Of primary importance to the calculation of runoff at depth is the calculation of the shock front. The program starts the shock wave when surface melt begins or changes from a decreasing to an increasing melt rate. The program then advances iteratively with a set time interval, calculates the slope of the shock at each point from eq. 3 and then calculates the depth of the shock at the next time interval. This procedure is repeated until the shock intercepts the depth of interest.

While difficult to do graphically, the computer can easily handle the interpolation to get precise values of $\underline{u_+}$ (boundary condition) and $u_-$ (initial condition) for any given time-depth (t,z) combination needed to satisfy eq. 3. Using eq. 2, the characteristics ($u_+$, $u_-$) which pass through any point (t,z) can be found by iteration, interpolation or a variety of methods. With the ability to find these characteristics at any point, generation of the shock front becomes relatively straightforward, using eq. 3 to find the slope of the shock at this point. Values of flux at depth prior to and following the time of intersection of the shock front with depth are calculated similarly. Before the shock, the initial conditions provide the flux values while the boundary conditions are used to generate the flux values after the shock, both in the same time-stepping manner using eq. 2.

Both programs were written in Fortran IV on the Dartmouth Time Sharing System (DTSS) which uses a Honeywell 66/40 computer. Some changes will be necessary when attempting to operate the programs on a different computer configuration. The changes are primarily in the input-output sections of the program and may be easily replaced with conventional I/O statements for batch processing.

Computation of Flux at Depth with Real Data. Calculating flux at depth with actual measured surface melting can be quite complex, especially in the situation where more than one shock wave is generated during a day. The program written to account for these cases, then, is quite complex and lengthy. Composed of a main program and 5 subroutines, it occupies about 16,000 words of core storage. Required program inputs are described in Appendix A. A listing of the program as it is run on the DTSS is in Appendix B.

Files (disc, tape, cards) containing the day's surface runoff (boundary condition) and that part of the previous day's runoff which will make up the initial conditions are read and stored. If a sufficient number of data points do not exist, subroutine MORPTS adds the necessary points by interpolation. This is especially important at small values of flux where interpolations can cause large errors. Since the start of a melting event normally begins with a shock front, the shock is calculated initially. Subroutine SHOCK, once given the slope of the shock from subroutine SLOPE for a t,z pair, calculates z for the next time t.

SLOPE finds $u_+$ for any desired point (t,z) by searching the boundary conditions and calculating for each input characteristic:

$$t_{u_+zi} = z \left.\frac{dz}{dt}\right|_{u_+i} + t_{u_+si} \qquad (4)$$

where $t_{u_+si}$ is the time that this characteristic ($u_+i$) leaves the surface, $\left.\frac{dz}{dt}\right|_{u_+i}$ is the slope of this characteristic from eq. 2, and $t_{u_+zi}$ is the time of intersection with z of this particular characteristic. When a pair is found such that $t_{u_+zi}$ is less than t and $t_{u_+zi+1}$ is greater than t, then $u_+(t,z)$ can be calculated from

$$u_+(t,z) = (t-t_{u_+zi}) \;/\; (t_{u_+zi+1}-t_{u_+zi}) \cdot (u_+i+1 - u_+i) + u_+i. \quad (5)$$

Similarly $u_-(t,z)$ is found by searching the initial conditions and calculating

$$z_{u_-ti} = (t-t_{u_-si}) \cdot \left.\frac{dz}{dt}\right|_{u_-i} + z_{u_-oi}. \quad (6)$$

Then

$$u_- = (z-z_{u_-ti})/(z_{u_-ti+1}-z_{u_-ti}) \cdot (u_-i+1-u_-i) + u_-i. \quad (7)$$

Figure 2 shows details of this procedure. The slope of the shock front at (t,z) is then calculated from eq. 3 and is passed to subroutine SLOPE. Two techniques for projecting the shock front to the next time interval, the simple Euler's method where the next depth increment is merely the product of slope and timestep, and a more complex technique, the Fourth Order Runge-Kutta method (Conte 1965) were tested and their results are reported in a later section. Once the shock front intersection time with z is found, values of u at that depth for a chosen time interval, are determined by subroutine GAPFIL using the respective initial and boundary conditions.

The starting point of the next shock (if any) is identified from the surface melt rate by a slope reversal from negative to positive. Initial and boundary conditions are established for this shock in the main program and the shock generating procedure is repeated. This time, however, a check is made to see if the second shock intercepts the previous shock. If so, subroutine NTRCPT is called to find the point of intersection of the two shock fronts. From this point, a new shock is begun, using initial conditions of the first shock and boundary conditions of the second shock. When the last shock intersection of the data section (1 day) is found, GAPFIL finds values of u at the selected depth until u falls below $10^{-6}$ m/s or until one full day since the last shock intersection has expired.

Limits on the program are quite constraining at present, primarily because of the small amount of core storage allowed on the DTSS. A strong recommendation is to increase the dimension lengths of all variables included in a DIMENSION or COMMON declaration. Time step sizes are presently very critical in this regard. A shock front is limited to a total of 200 steps (120,000 s at a 600 s step size), while the total u output at depth is limited to 300 values (180,000 s at 600 s step size). A late arrival of a shock could, therefore, cause the program to "bomb-out" if it attempts to run 1 day past the last shock. The number of input initial condition values is limited to 120 and the surface runoff is limited to 200 (u,t) pairs.

If the program is to be used for more specialized purposes, parts can be deleted or restructured with little difficulty. For instance, if a case of intersecting shock waves will never occur, subroutine NTRCPT and a part of the main program can be deleted. Some smoothing of the input is recommended in order to reduce the number of shocks. Physically, very small shocks will be wiped out and absorbed rather quickly by larger shocks in any case. The user must test the abilities of the program to adapt to his situation and modify it accordingly.

Approximation of Surface Flux with a Function. In many cases it is desirable to simply approximate the surface runoff by some relatively simple function rather than a detailed complicated input. This is especially true in cases where multilayered snowpack behavior is being simulated or in a situation of strong radiative melting where the actual melt can be very closely approximated by a function (Colbeck and Davidson 1973). The program that accommodates this general case is somewhat more streamlined and efficient than that described previously. This program consists of a main program, 2 subroutines and 3 function subprograms. Program inputs and complete listings are included in Appendices C and D.

Surface melt is assumed to occur in half day (0-43200 s) and the initial conditions (if any) are generated by surface runoff occurring in the same time period of the previous day. These conditions are controlled by the maximum surface flux (Umax), the snow properties ($k^{1/3}/\phi_e$), and the function governing the runoff profile (presently a sinewave).

Although different in many respects from the previous program, the primary difference is in the calculation of the $u_+$ and $u_-$ for a given (t,z) pair. Two subroutines, UPLUS and UMINUS, find $u_+$ and $u_-$ respectively using an iterative technique. Input to the subroutines are (t,z) and the limits of a search interval ($t_1$, $t_2$) established by the last call to the subroutine. These subroutines use functions FNU and FNZ for the search procedure. FNU is the function that determines u for a given input time using a chosen mathematical function (sine function in this case). FNZ generates a depth (z) for an input u and time it left the surface ($t_{usi}$) using eq. 6.

An error condition E controls the iterative search:

$$E = 1 - \left(\left.\frac{dz}{dt}\right|_u \cdot (t-t_{usi})\right)/z \qquad (8)$$

where $t_{usi}$ is given by $t_{usi} = (t_1+t_2)/2$ and the quantity $\left(\left.\frac{dz}{dt}\right|_u \cdot (t-t_{usi})\right)$ is provided by function FNZ after the $u_i$ for $t_{usi}$ is generated by FNU. The time t is obtained by time stepping as in the previous program. If the value E is less than 0.002 (arbitrary criterion), then the $u_i$ having $t_{usi}$ as its surface start time is selected as that passing through (t,z). If the criterion is not met, $t_{usi}$ is changed by assigning the value of $t_{usi}$ to $t_1$ or $t_2$ (depending on the sign of E) and recomputing

$t_{usi} = (t_1+t_2)/2$. The iteration continues until the E criterion is met, usually in less than 20 iterations.

When $u_+$ and $u_-$ for the given (t,z) have been calculated, the slope of the shock is calculated using function FNS and the next depth is computed from Euler's technique. Once the shock intercept with depth is calculated, values of u before and after the intercept at a chosen time interval are obtained by calling the applicable UMINUS or UPLUS subroutines at each interval.

This program is also written in FORTRAN IV and should be adaptable to most modern computer systems with little difficulty. A feature that may prove useful is that the function of time that describes the surface flux may be easily changed. If some other function, say a polynomial, better fits a certain melting situation, it requires only that the function program FNU be modified. If that function occurs over some time interval other than the standard 43200s, other parts of the program must be changed.

Test Cases and Results.

Single Peak Input. A surface melting profile containing one single peak was used for a rigorous error analysis of the first program. Figure 3 shows the input plus the output generated by the program for a one day period. Error in all test cases was calculated with a planimeter; assuming the conservation of liquid mass, the area under the curves must be equal. The output will be greater or less than the input if the shock front intercept is early or late, respectively. Measurement error with the planimeter is on the order of 1%.

Table I gives computer time (including compilation time) and output error for various time step intervals for both the Euler's and the Runge-Kutta (RK) Fourth Order methods of determining shock penetration. The single-peak case considered had a Umax of $1.59 \times 10^{-6}$ m/s, $k^{1/3}/\phi_e$ of 0.00178 $m^{2/3}$ and depth of 1.25 m. It is interesting to note that, while the RK method yields a fairly consistent positive error (shock intercepting too early) regardless of the time interval, Euler's technique yields errors that vary with time step, going negative (shock too late) as the interval becomes too coarse. In both cases the positive error is believed to be caused by round off error, accumulating as the number of time steps increases, and by the inability to interpolate accurately the very small values of the initial conditions near start time. The optimum shock front time interval for this case appears to be 600-900 s. Nothing seems to be gained by using the RK method over Euler's as computer time and error are both greater for the RK method.

Table II shows the Euler's method applied to 3 other single-peak input cases, all having a Umax of $1.59 \times 10^{-6}$ m/s, depth z of 2.05 m, and having different $k^{1/3}/\phi_e$ for the same time step sizes. All cases show that surprisingly large time intervals yield the best results. Figure 4 shows the number of time steps required for the shock front to intersect the chosen depth for the previous 4 cases. It appears that as the program exists now, something between 22 and 35 steps is optimum; that is, the program should be run initially with any step size to determine the approximate shock front intersection time. Then this time divided by say 25 should result in a fairly optimum time step interval. In cases of multiple shock fronts this procedure should apply to the first shock intersection with depth. If no initial conditions are used, it is recommended that a time step of 600 s or less be used.

Similar tests were made with the function input program, in all cases using one-half a wavelength of a sine wave with a period of 86,400 s. Table III gives time, step size, computer time and output error for 4 different combinations of depth, $k^{1/3}/\phi_e$ and Umax. Errors are considerably less in this program, probably because linear interpolation is not necessary when finding a particular $u_+$ or $u_-$. The error versus time step interval from Table III are plotted in Figure 5. This Figure shows that a time step size of 600 to 900 s is optimum for the cases shown, independent of depth and snow properties.

Multipeak Input. The surface flux of water is often characterized by multipeak inputs because of variable rainfall intensities and/or varying atmospheric conditions. The occurrence of multiple maximums introduces problems in the construction of the flow field because a new shock front is generated at the surface each time the surface flux stops decreasing and increases. These multiple shocks are handled by the program as illustrated on Figure 6 for the double-peaked input. This particular example illustrates the dynamics of flow through unsaturated snow. While the input is symmetrical, with increasing depth the flow is increasingly skewed towards larger times. The first peak of the input is partially eroded away at a depth of 1 m, but the second peak still retains its full value. The reason is that, while the first peak has been overtaken by a shock front, the second peak is still moving along its own characteristic. By 2-m depth, the first peak has almost completely disappeared and the second peak is almost overcome by the second shock as evidenced by the expanding vertical line just below the second peak. At 3-m depth, the first peak has disappeared entirely and the second peak has been partially absorbed by the second shock. At greater depths, the maximum flux decreases, the minimum flux increases and the peak shifts to later times just as for a single-peaked input. Clearly the maximum effect of the multiple-peaked inputs occurs at shallow depths. When only small variations occur in an otherwise smooth surface input, the effects of these perturbations damp out with depth very quickly and they have no significant effect on the flow field.

Skewed Input. The value of this computer program as a research tool is illustrated by Figure 7 which shows the movement of symmetrical and skewed inputs of the same duration, volume and peak. The symmetrical input represents surface melting simulated by a sinusoidal function, and the skewed input represents surface melting which peaks late in the afternoon rather than in the middle of the day. While this is an extreme case of skewed surface flux, it is important to test the assumption that clear weather melting can be simulated by a symmetrical function (Colbeck and Davidson 1973).

The flow at 2-m depth is significantly affected by the skew, although the peaks are separated by less than the 3-hour difference at the surface. The major difference at 2 m is that the shock front has just reached the peak of the symmetrical input but has not yet reached the peak of the skewed input. At 4-m depth, both peaks have been eroded significantly by the shock front and the difference between the peaks has been reduced by about 60%. The difference between the peaks continues to disappear with increasing depth because the shock from the skewed input arrives later but moves faster since it has a greater strength (i.e., $u_+ - u_-$). At 8-m depth, the maximum value of flux is just over one-half of its original value and the distance between the peaks is only one-fifth of the spacing of the input peaks.

The difference between these two inputs may be significant at shallow depths but the skewed input used here is an extreme case of melt shifted to the late afternoon. Shifts of 1/2 to 1-hour are common but would not introduce large errors in the calculated peak flow rate or lag time. Since the error is dependent on snow depth, snow properties, peak flux and phase shift, each individual will have to decide if the simple sinusoidal function is sufficiently accurate for his purposes.

Conclusions.

The availability of this computer program satisfies the need of researchers who have been laboriously constructing the characteristics and shock fronts by hand (e.g., Dunne et al. 1976). There are many possible research applications of this program including a complete investigation of the effect of skewed inputs, sensitivity analyses of the effects of grain size, and density and layering. These requirements can all be satisfied by use of part or all of the program. Unfortunately, the complete program may be too long for the practical purposes of hydrological forecasting. In Anderson's (1973) model, for example, the program would replace a relationship between lag and excess water. This relationship, which is very similar to eq. 2, works well over time periods of 6-hours for shallow snowcovers, but would be inappropriate for shorter time periods or deeper snowcovers where the dynamics of the intersecting characteristics would control the timing of the water runoff. Those responsible for constructing forecasting models will have to decide if the increased computer time is justified by the increased accuracy and sensitivity to the input parameters.

## Literature Cited


Anderson, E.A. (1973). National Weather Service river forecast system - Snow accumulation and ablation model, National Oceanic and Atmospheric Aministration, National Weather Service, Technical Memorandum, NWS-HYDRO-17.

Colbeck, S.C. (In press). The physical aspects of water flow through snow. Advances in Hydroscience. New York: Academic Press.

Colbeck, S.C. and G. Davidson (1973). Water percolation through homogeneous snow. In The Role of Snow and Ice in Hydrology. Proceedings of Banff Symposia, September, 1972, UNESCO-WMO-International Association of Scientific Hydrology, Paris-Geneva-Budapest, vol. 1, p. 242-57.

Conte, S.D. (1965). Elementary Numerical Analysis. New York: McGraw-Hill.

Corps of Engineers (1956). Summary report of snow investigations. Snow Hydrology. U.S. Army, Corps of Engineers, North Pacific Division, Portland, Oregon.

Dunne, T., A.G. Price and S.C. Colbeck (1976). The generation of runoff from subarctic snowpacks. Water Resources Research, vol. 12, no. 4, p. 677-85.

U.S. Army Engineering Division (1972). Program description and user manual for SSARR. Streamflow synthesis and reservoir regulation. U.S. Army, Corps of Engineers, North Pacific Division, Portland, Oregon.

Computer time and output error for Euler's and Runge-Kutta methods of determining shock penetration.

Table I

Single Peak, Umax = $1.59 \times 10^{-6}$ m/s, $k^{1/3}/\phi_e = 0.00178\ m^{2/3}$, $z = 1.25$ m

| Technique | Time step (s) | Computer time (s) | Output error (%) |
|---|---|---|---|
| RK | 600 | 14.8 | 1.3 |
| Euler | 600 | 6.2 | 0.8 |
| RK | 900 | 11.1 | 1.3 |
| Euler | 900 | 5.1 | 0.3 |
| RK | 1200 | 9.1 | 1.3 |
| Euler | 1200 | 4.8 | -0.2 |
| RK | 1500 | 7.8 | 1.3 |
| Euler | 1500 | 4.3 | -0.7 |
| RK | 2000 | 6.5 | 1.2 |
| Euler | 2000 | 4.1 | -1.4 |

Euler's method applied to 3 single-peak cases.

Table II

Single Peak, Umax = $1.59 \times 10^{-6}$ m/s, z = 2.05 m

| $k^{1/3}/\phi_e$ ($m^{2/3}$) | Time step (s) | Computer time (s) | Output error (%) |
|---|---|---|---|
| 0.00178 | 300 | 10.7 | 2.3 |
| " | 600 | 7.0 | 1.7 |
| " | 900 | 5.9 | 1.1 |
| " | 1200 | 5.2 | 0.5 |
| " | 1500 | 4.9 | -0.1 |
| " | 2000 | 4.7 | -0.9 |
| 0.00356 | 300 | 7.2 | 1.2 |
| " | 600 | 5.0 | 0.8 |
| " | 900 | 4.2 | 0.4 |
| " | 1200 | 4.1 | 0.0 |
| " | 1500 | 3.7 | -0.5 |
| " | 2000 | 3.4 | -1.4 |
| 0.00089 | 300 | 12.3 | 2.3 |
| " | 600 | 10.9 | 2.0 |
| " | 900 | 8.4 | 1.6 |
| " | 1200 | 7.1 | 1.2 |
| " | 1500 | 6.3 | 0.9 |
| " | 2000 | 5.3 | 0.5 |

Time, step size, computer time and output error
for 4 different cases.

Table III

Sine wave function

| Umax (m/s) | $k^{1/3}/\phi_e$ ($m^{2/3}$) | Depth (m) | Time step (s) | Computer time (s) | Output error (%) |
|---|---|---|---|---|---|
| $1.59 \times 10^{-6}$ | 0.00178 | 2.05 | 300 | 8.4 | 0.9 |
| " | " | " | 600 | 6.1 | 0.3 |
| " | " | " | 900 | 5.1 | -0.3 |
| " | " | " | 1200 | 4.3 | -0.9 |
| " | " | " | 1500 | 4.3 | -1.5 |
| " | " | " | 2000 | 4.3 | -2.1 |
| $1.25 \times 10^{-6}$ | 0.00159 | 3.15 | 150 | 21.7 | 0.9 |
| " | " | " | 300 | 12.5 | 0.5 |
| " | " | " | 600 | 7.5 | 0.0 |
| " | " | " | 900 | 6.1 | -0.3 |
| " | " | " | 1200 | 5.3 | -1.0 |
| " | " | " | 1500 | 5.1 | -1.7 |
| " | " | " | 2000 | 4.3 | -1.7 |
| " | " | 1.50 | 150 | 13.7 | 1.2 |
| " | " | " | 300 | 8.2 | 0.8 |
| " | " | " | 600 | 5.8 | 0.3 |
| " | " | " | 900 | 5.1 | -0.1 |
| " | " | " | 1200 | 4.6 | -0.5 |
| " | " | " | 1500 | 4.5 | -0.9 |
| " | " | " | 2000 | 4.3 | -1.5 |
| " | 0.00308 | 3.15 | 150 | 14.8 | 1.1 |
| " | " | " | 300 | 8.6 | 0.6 |
| " | " | " | 600 | 6.1 | 0.2 |
| " | " | " | 900 | 5.1 | -0.2 |
| " | " | " | 1200 | 4.6 | -0.6 |
| " | " | " | 1500 | 4.4 | -1.1 |
| " | " | " | 2000 | 4.1 | -1.7 |

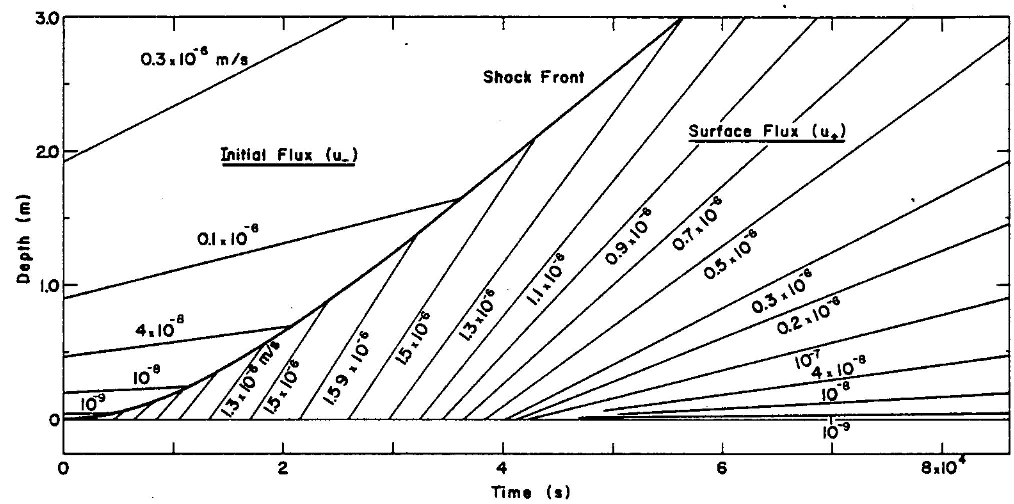


Figure 1. The characteristics and shock front for a typical day of clear weather melting. The surface melting, or boundary condition, is a sinusoidal input with an amplitude of $1.59 \times 10^{-6}$ $m^3/m^2/s$ and a duration of 43,200 s (12h). The initial condition, or antecedent flow, forms the intercepts on the $\underline{z}$ axis and is taken from the trailing edge of melting on the previous day. The snow properties are characterized by setting $k^{1/3}/\phi_e^{-1}$ equal to 0.00178 $m^{2/3}$. The larger values of flux ($u_+$) from the current day and smaller values of flux from the previous day ($u_-$) join at the shock front according to eq. 3. The slope of each characteristic representing values of $\underline{u}$ is given by eq. 2.

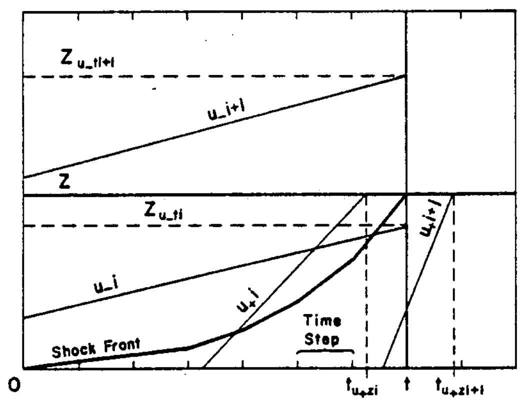


Figure 2. Details of the computer technique for finding $u_+$ and $u_-$ at a specific time and depth (t,z) along the shock front.

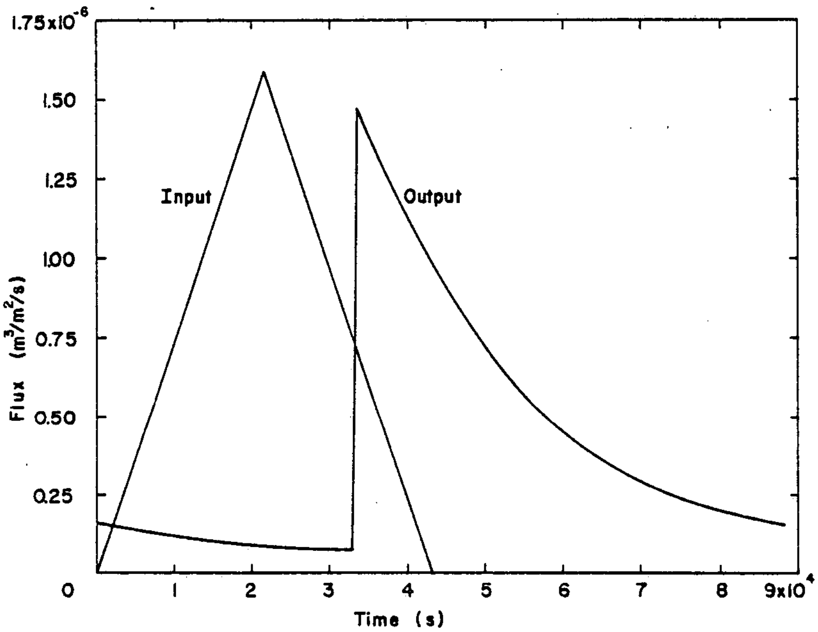


Figure 3. A typical input profile used for test cases and the resulting one-day runoff profile at a depth of 2.05 m.

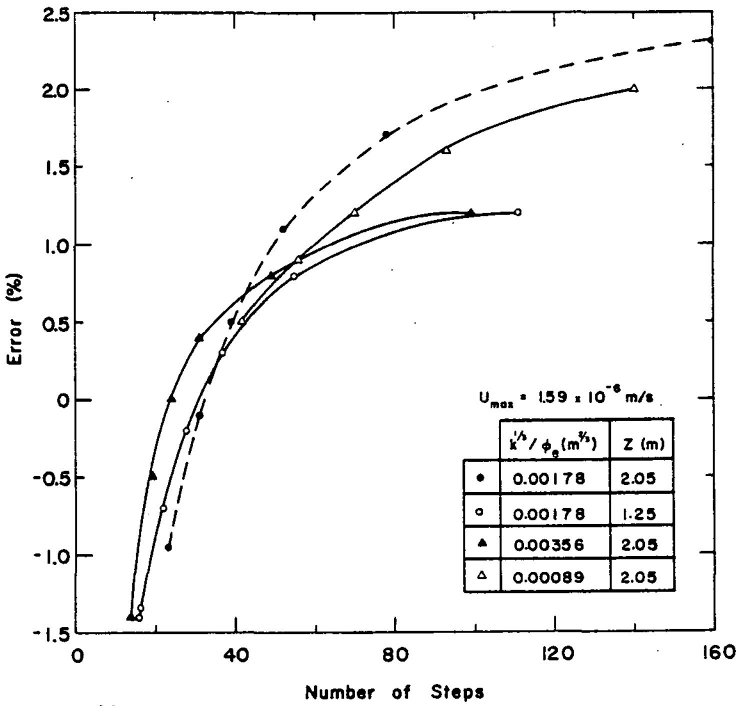


Figure 4. Error versus time steps required for shock front intersection with depth for the single peaked linear input using Euler's technique.

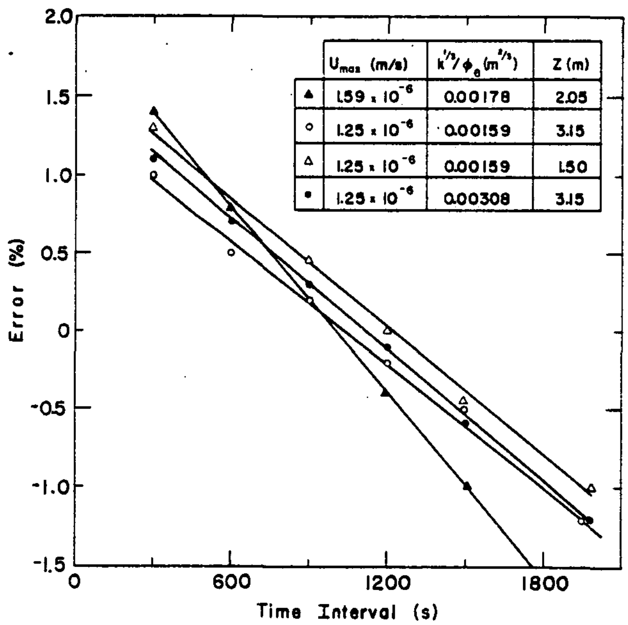


Figure 5. Error versus time step interval for sine wave inputs with varying depths, $k^{1/3}\phi_e$ and flux magnitudes.

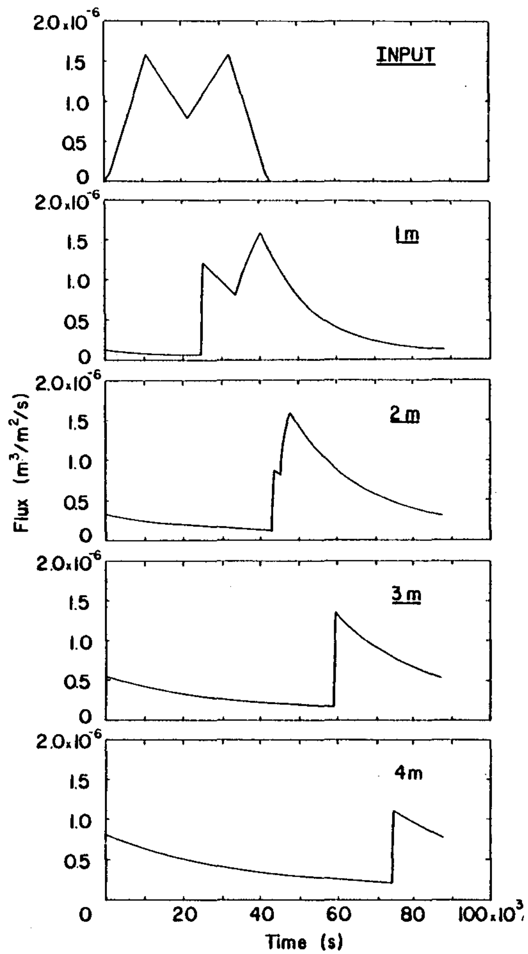


Figure 6. A double-peaked input with a duration of 43,200 s (12h) and a maximum of $m^3/m^2/s$ moves through snow where $k^{1\ 3}\ \phi_e = 0.00178$ $m^2/^3$. The flow versus time at four depths is shown.

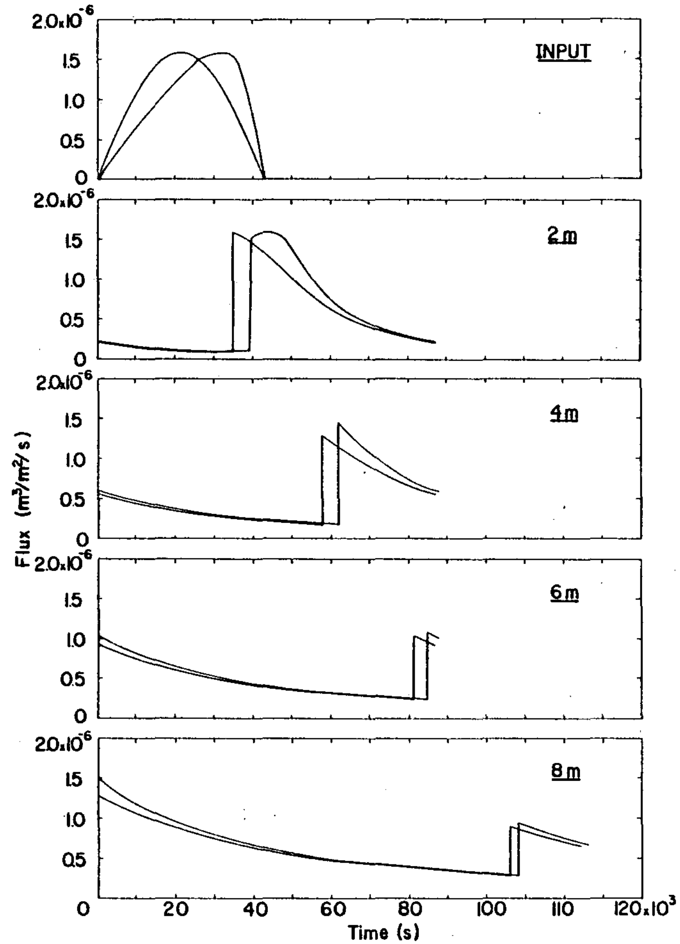


Figure 7. The flow at four depths is shown for a symmetrical, sinusoidal input and a skewed input which peaks 3h later. The two inputs have the same peak, period and volume. The difference between the outputs disappears with increasing depth.

## Appendix A: Input Parameters for Program for Actual Data

1. Initial Condition Flag (L1) - parameter to indicate whether or not there are initial conditions; 1 = yes, 0 = no.
2. Initial Condition Filename (FN1) - file on which initial condition flux values and time ($u_-$,t) are stored.
3. Surface Flux Filename (FN2) - file where surface flux values ($u_+$,t) are stored.
4. Output Filename (FN3) - file that flux values and time (u,t) at depth are to be written to.
5. Shock Wave Time Step (H) - time step (sec) used for generating the shock front.
6. Time Step at Depth (H2) - interval (sec) that values are to be generated at depth (normally 600 sec).
7. Depth (DD) - depth (cm) of interest in the snowpack.
8. $k^{1/3}/\phi_e$ (C2) - snow property parameter ($cm^{2/3}$).

### Comments on Inputs

All files have flux data in the sequence $u_1$, $t_1$, $u_2$, $t_2$, ..., $u_i$, $t_i$ and the files can be easily changed to cards, tape or other mass storage devices. Most critical of the inputs is the initial conditon data set. **These data should consist of the last major negative slope of the previous day's input with no slope reversals included.** Also note that all units in the program are centimeter-gram-second.

APPENDIX B

SNOWFLUX, Computer Program for actual Runoff Data

```
SNOFLUX

*     PROGRAM TO COMPUTE FLUX AT DEPTH OF A SNOWPACK,CAPABLE OF HANDLING
*     MULTIPLE SHOCK WAVES, ASSUMES BOUNDARY AND INITIAL CONDITIONS
*     ARE LINEAR
*     F1,FT1,F2,FT2,ARE INITIAL CONDITIONS;B,BT=BOUNDARY CONDITIONS
*     U,T=ENTIRE INPUT BOUN. COND.;TD,UD=FINAL U,T AT DEPTH
*     TS,DS=PATH OF LAST GOOD SHOCK,ST1,SD1,ST2,SD2=PATHS OF 2 SHOCKS
*     SAVED TO FIND INTERSECTION
      LIBRARY "EULER"
      CHARACTER FN1*8,FN2*8,FN3*8
      DIMENSION F1(120),F2(120),FT1(120),FT2(120),B(120),BT(120)
      DIMENSION U(200),T(200)
      COMMON/BLK3/UD(300)
      COMMON/BLK1/DS(200),TS(200)/BLK/SD1(200),SD2(200),ST1(200),ST2(200)
*     ARE FIRST SET OF INITIAL CONDITIONS FROM SEPARATE FILE
      PRINT,"INITIAL CONDITION FILE YES-1,NO-0"
      INPUT,L1
      IF(L1.EQ.0)GO TO 20
      PRINT,"INITIAL CONDITION FILE"
      INPUT,FN1
      OPENFILE 1,FN1,"NUMERIC"
      DO 10 I=1,2000
      NF2=I
      READ(1,END=15)F2(I),FT2(I)
      F1(I)=F2(I)
      FT1(I)=FT2(I)
   10 CONTINUE
*     CHECK FOR ENOUGH POINTS,SET MINIMUM IS 1 EVERY 500 SEC FOR INITIAL
   15 NF2=NF2-1
      FIN=FT2(NF2)-FT2(1)
      FRATIO=FIN/NF2
      KCALL=0
      IF(FRATIO.LT.500)GO TO 25
      CALL MORPTS(F2,FT2,U,T,NF2,KCALL)
      DO 12 I=1,NF2
      F1(I)=U(I)
      F2(I)=U(I)
      FT1(I)=T(I)
      FT2(I)=T(I)
   12 CONTINUE
      GO TO 25
   20 NF2=0
   25 PRINT,"SURFLUX FILE,OUTPUT FILE"
      INPUT,FN2,FN3
      OPENFILE 2,FN2,"NUMERIC"
      OPENFILE 3,FN3,"NUMERIC"
*     READ IN BOUNDARY CONDITIONS
      DO 30 I=1,2000
      NUM=I
      READ(2,END=35)U(I),T(I)
   30 CONTINUE
```

SNOFLUX (continued)

```
   35 NUM=NUM-1
*     CHECK FOR A POINT EVERY 800 SEC FOR BOUNDARY CONDITIONS
      TINT=T(NUM)-T(1)
      PRATIO=TINT/NUM
      KCALL=1
      IF(PRATIO.GT.800)CALL MORPTS(B,BT,U,T,NUM,KCALL)
      PRINT,"SHOCK WAVE STEP,INTERPOLATION STEP"
      INPUT,H,H2
      PRINT,"DEPTH,K**(1/3)/PHI-E"
      INPUT,DD,C2
      C1=(54700.**(1./3.)*C2)
      CC=3.*C1
      K=1
      TL=T(1)
      N=1
*     EVERYTHING ENTERED,PROCEED THRU DATA,GO TO BOUNDARY CONDITION SETUP
      KFLG =0
      KEND=0
      GO TO 125
*     SET UP INITIAL CONDITIONS
   50 DO 100 I=N,NUM
      L=I
      IF(U(I+1).LT.U(I))GO TO 110
  100 CONTINUE
*     HAVE START OF INITIAL COND.,PUT THEM IN TEMP STORAGE
*     USE 2 SETS OF INITIAL COND.:F1,FT1 ARE FROM LAST SHOCK HOLD IN
*     CASE OF INTERSECTION OF SHOCKS
  110 M=0
      DO 130 I=L,NUM
      M=M+1
      K=I
      F2(M)=U(I)
      FT2(M)=T(I)
      IF(U(I+1).GT.U(I))GO TO 120
  130 CONTINUE
*     SET UP UPSLOPE OF BOUNDARY CONDITIONS
  120 NF2=M
      N=K
  125 M=0
      KI1=L
      DO 150 I=K,NUM
      M=M+1
      KI=I
      B(M)=U(I)
      BT(M)=T(I)
      IF(U(I+1).LT.U(I))GO TO 140
  150 CONTINUE
*     ESTABLISH BEGINNING POINT FOR NEXT SET OF INITIAL CONDITIONS
*     SET UP DOWNSLOPE OF BOUNDARY CONDITIONS
  140 L=KI
```

SNOFLUX (continued)

```
      DO 170 I=KI+1,NUM
      M=M+1
      B(M)=U(I)
      BT(M)=T(I)
      IF((I+1).GT.NUM)GO TO 175
      IF(U(I+1).GT.U(I))GO TO 180
 170  CONTINUE
 175  KEND=1
 180  NB=M
*     FIND PATH OF SHOCK STARTING FROM SURFACE
      TB=BT(1)
      DI=DD
      CALL SHOCK(0.,TB,NB,B,BT,NF2,F2,FT2,C1,CC,DI,TI,H,NPTS2)
      PRINT,"SHOCK FROM SURFACE,TIME,DEPTH",TI,DI
      IF(KFLG.EQ.1)GO TO 250
      IF(TI.LT.TL)GO TO 250
      DO 190 I=1,NF2
      F1(I)=F2(I)
      FT1(I)=FT2(I)
 190  CONTINUE
      NF=NF2
      KII=KI1
      IF(DI.LT.DD)GO TO 200
      KFLG=0
*     INTERPLOATE U'S AT DEPTH BETWEEN SHOCKS
      CALL GAPFIL(NF2,F2,FT2,TL,TI,U,T,DD,CC,KEND,KII,H2)
      GO TO 300
*     COME HERE IF DEPTH NOT ACHIEVED ON SHOCK,MEANS NEXT MUST INTERCEPT
 200  KFLG=1
      GO TO 300
 250  DO 280 I=1,NPTS2
      SD2(I)=DS(I)
      ST2(I)=TS(I)
 280  CONTINUE
*     FIND INTERCEPT AND CONTINUING SHOCK WAVE
      CALL NTRCPT(NPTS1,NPTS2,H,T1,D1)
*     FIND CONTINUATION SHOCK PATH AFTER INTERSECTION
      CALL SHOCK(D1,T1,NB,B,BT,NF,F1,FT1,C1,CC,DI,TI,H,NPTS2)
      PRINT,"SECONDARY SHOCK TIME,DEPTH",TI,DI
      IF(DI.LT.DD)GO TO 200
      KFLG=0
*     MAKE THIS A GOOD SHOCK
      IF(TI.LT.TL)GO TO 300
      CALL GAPFIL(NF,F1,FT1,TL,TI,U,T,DD,CC,KEND,KII,H2)
 300  DO 350 I=1,NPTS2
      SD1(I)=DS(I)
      ST1(I)=TS(I)
 350  CONTINUE
      NPTS1=NPTS2
      IF(KFLG.EQ.0)TL=TI
```

```
      IF(KEND.EQ.0)GO TO 50
*     THIS SECTION FOR INTERPOLATION BETWEEN LAST SHOCK AND CUTOFF U
*     CARRY OUT NO LONGER THAN 2 DAYS BEYOND LAST SHOCK IF U NOT REACHED
      TI=TL+86400
      CALL GAPFIL(NB,B,BT,TL,TI,U,T,DD,CC,KEND,L,H2)
      DO 900 I=1,KEND
*     OUTPUT FINAL U'S,T'S AT DEPTH
      TD=I*H2+T(1)
       WRITE(3)UD(I),TD
  900 CONTINUE
      PRINT,"DISCONTINUED CALCULATIONS AT TIME,U=",TD,UD(KEND)
      TO=TD-DD/(CC*UD(KEND)**(2./3.))
      REM=(DD/(SQRT(54700.)))*(DD/3.)*(C2**(3./2.))/(SQRT(TD-TO))
      PRINT,"REMAINDER = ",REM
      STOP
      END
      SUBROUTINE SHOCK(Z,T,NB,UB,UT,NI,VI,VT,C1,CC,DD,TI,H,NPTS)
*     FINDS SHOCK WAVE BEGINNING AT ANY TIME AND DEPTH TO DESIRED DEPTH
*     ASSUMES THAT IT HAS NECESSARY INITIAL AND BOUNDARY CONDITIONS
      DIMENSION UB(120),UT(120),VI(120),VT(120)
      COMMON/BLK1/DS(200),TS(200)
      REAL K1,K2,K3,K4
      K=0
      TI=T
      IF(Z.EQ.0.)GO TO 25
*       SAVE PREVIOUS SHOCK PATH IF THIS IS A CONTINUATION
      DO 30 I=1,2000
      K=I
      IF(DS(K).GT.Z)GO TO 40
   30 CONTINUE
   40 DS(K)=Z
      TS(K)=T
*     BEGIN EULER'S METHOD
   25 D1=Z
      CALL SLOPE(TI,Z,NB,UB,UT,NI,VI,VT,C1,CC,SS,KEXC)
      Z=Z+H*SS
      TI=TI+H
      K=K+1
      TS(K)=TI
      DS(K)=Z
      IF(K.EQ.1)GO TO 50
      IF(KEXC.EQ.0)GO TO 50
      DD=DS(K)
      TI=TS(K)
      NPTS=K-1
      GO TO 100
   50 IF(Z.LT.DD)GO TO 25
      TI=((DD-D1)/(Z-D1))*(TI-(TI-H))+(TI-H)
      NPTS=K
  100 RETURN
```

```
      END
      SUBROUTINE SLOPE(T,Z,NB,UBOUN,TBOUN,NI,UINIT,TINIT,C1,CC,SS,KEXC)
      DIMENSION UBOUN(120),TBOUN(120),UINIT(120),TINIT(120)
      DIMENSION TS(120),ZK(120)
*     FINDS THE SLOPE OF THE SHOCK AT A GIVEN TIME AND DEPTH
      KEXC=0
      IF(UBOUN(1).GT.0.)GO TO 5
      TS(1)=TBOUN(1)
      GO TO 7
    5 TS(1)=TBOUN(1)+Z/(CC*UBOUN(1)**(2./3.))
*     LOOP FINDS CHARACTERISTICS BEFORE AND AFTER T AT Z DEPTH (U+)
    7 DO 10 I=2,NB
      IF(UBOUN(I).GT.0.)GO TO 8
      TS(I)=TBOUN(I)
      GO TO 9
    8 TS(I)=Z/(CC*UBOUN(I)**(2./3.))+TBOUN(I)
     9 IF(TS(I).GT.T.AND.TS(I-1).LE.T)GO TO 20
      GO TO 10
   20 UPLUS=(T-TS(I-1))/(TS(I)-TS(I-1))*(UBOUN(I)-UBOUN(I-1))+UBOUN(I-1)
*     CHECK FOR BOUNDARY CONDITIONS TOO LOW OR TOO HIGH
      IF(TS(I-1).GT.0.)GO TO 25
   10 CONTINUE
      IF(T.GT.TS(NB))GO TO 22
      UPLUS=UBOUN(1)
      GO TO 25
   22 KEXC=1
   25 IF(NI.GT.0)GO TO 35
      UMINUS=0.
      GO TO 65
    35 ZK(1)=(T-TINIT(1))*(CC*UINIT(1)**(2./3.))
*     LOOP FINDS CHARACTERISTICS ABOVE AND BELOW Z AT TIME T (U-)
      DO 50 I=2,NI
      K=I
      ZK(I)=(T-TINIT(I))*(CC*UINIT(I)**(2./3.))
      IF(ZK(I).LE.Z.AND.ZK(I-1).GT.Z)GO TO 60
   50 CONTINUE
      UMINUS=UINIT(K)
      GO TO 65
   60 UMINUS=(Z-ZK(K))/(ZK(K-1)-ZK(K))*(UINIT(K-1)-UINIT(K))+UINIT(K)
*     SS IS SLOPE OF SHOCK AT T,Z
   65 SS=C1*(UPLUS**(2./3.)+UPLUS**(1./3.)*UMINUS**(1./3.)+UMINUS**(2./3.))
   79 RETURN
      END
      SUBROUTINE NTRCPT(NPTS1,NPTS2,H,TI,DI)
      COMMON/BLK/SD1(200),SD2(200),ST1(200),ST2(200)
*     ROUTINE TO FIND INTERCEPT OF 2 SHOCK WAVES
*     MOVE RAPIDLY THRU FIRST POINTS
      DO 50 I=1,NPTS1
      L=I
      IF(ST1(I).GT.ST2(1))GO TO 75
```

SNOFLUX (continued)

```
   50 CONTINUE
   75 DO 100 I=L,NPTS1
      DO 150 K=1,NPTS2
      M=I
      N=K
      IF(ST2(K).LT.ST1(I).AND.SD2(K).GE.SD1(I))GO TO 200
  150 CONTINUE
  100 CONTINUE
  200 I=M-2
      K=N-2
*     CLOSED FORM SOLUTION OF INTERSECTING LINES
      S1=(SD1(M)-SD1(I))/(ST1(M)-ST1(I))
      S2=(SD2(N)-SD2(K))/(ST2(N)-ST2(K))
      FK1=SD1(I)-(S1*ST1(I))
      FK2=SD2(K)-(S2*ST2(K))
      TI=(FK1-FK2)/(S2-S1)
      DI=S1*(TI-ST1(I))+SD1(I)
      PRINT,"INTERSECTION OF 2 PREVIOUS SHOCKS",TI,DI
      RETURN
      END
      SUBROUTINE GAPFIL(NFF,FF,FFT,TL,TI,U,T,DD,CC,KEND,KII,H2)
      DIMENSION FF(120),FFT(120),U(200),T(200),DF(120),DFT(120)
      COMMON/BLK3/UD(300)
*     INTERPOLATES U'S BETWEEN LAST SHOCK TL,AND THIS SHOCK TI
      K=1
      IF(NFF.EQ.0)GO TO 300
      TB=DD/(CC*FF(1)**(2./3.))+FFT(1)
      IF(TB.LT.TL)GO TO 25
*     BYPASS INITIAL SEARCH IF THIS IS FIRST SHOCK
      IF(FFT(1).LE.T(1))GO TO 25
*      BYPASS INITIAL SEARCH IF THIS IS LAST SHOCK
      IF(TI.GT.TL+86200)GO TO 25
      DO 15 I=1,20
      M=KII-I
      TT=DD/(CC*U(M)**(2./3.))+T(M)
      IF(TT.LT.TL)GO TO 20
*     GO 20 POINTS BACK FROM BEGINNING OF INITIAL CONDITIONS IF NECESSARY
   15 CONTINUE
      PRINT,"ERROR,MUST GO BACK FURTHER THAN 20 PTS TO FILL GAP BET. SHOCKS"
   20 K=1
      KI1=KII-1
      DO 50 I=M,KI1
      DF(K)=U(I)
      DFT(K)=T(I)
      K=K+1
   50 CONTINUE
   25 DO 27 I=1,NFF
      DF(K)=FF(I)
      DFT(K)=FFT(I)
      K=K+1
```

SNOFLUX (continued)

```
   27 CONTINUE
      NF=IFIX((TL-T(1))/H2)+1
      NN=IFIX((TI-T(1))/H2)
      TB=DD/(CC*DF(1)**(2./3.))+DFT(1)
      TA=DD/(CC*DF(2)**(2./3.))+DFT(2)
      M=2
      DO 100 I=NF,NN
      L=I
      TD=I*H2+T(1)
  150 IF(TA.GT.TD.AND.TB.LT.TD)GO TO 200
      M=M+1
      TB=TA
      TA=DD/(CC*DF(M)**(2./3.))+DFT(M)
      GO TO 150
  200 UD(I)=((TD-TB)/(TA-TB))*(DF(M)-DF(M-1))+DF(M-1)
      IF(UD(I).LT.1.E-6)GO TO 700
  100 CONTINUE
*     STOP IF TWO DAYS PAST LAST SHOCK
      IF(TD.LT.(TL+85000))GO TO 800
  700 KEND=L
      GO TO 800
  300 INF=IFIX((TI-TL)/H2)+1
      DO 400 I=1,INF
      UD(I)=0
  400 CONTINUE
  800 RETURN
      END
      SUBROUTINE MORPTS(B,BT,U,T,NUM,KCALL)
*     ROUTINE TO INTERPOLATE ADDITIONAL POINTS ON BOUNDARY AND INITIAL
*     CONDITIONS
      DIMENSION B(120),BT(120),U(200),T(200)
      ISEC=360
      IF(KCALL.EQ.0)GO TO 90
      ISEC=720
      DO 120 I=1,NUM
      B(I)=U(I)
      BT(I)=T(I)
  120 CONTINUE
   90 M=1
      K=2
      U(1)=B(1)
      T(1)=BT(1)
      T2=BT(1)+ISEC
      NEED=IFIX((T(NUM)-T(1))/ISEC)+NUM
      DO 180 I=2,NEED
   80 IF(T2.LE.BT(M+1).AND.T2.GT.BT(M))GO TO 75
      M=M+1
      U(I)=B(M)
      T(I)=BT(M)
      GO TO 160
```

SNOFLUX (continued)

```
 75 U(I)=((T2-BT(M))/(BT(M+1)-BT(M)))*(B(M+1)-B(M))+B(M)
    T(I)=T2
    IF(T2.EQ.BT(M+1))M=M+1
    T2=T2+ISEC
160 IF(T2.GT.BT(NUM))GO TO 170
    K=K+1
180 CONTINUE
170 NUM=K
    RETURN
    END
```

Configuration of Subroutine SHOCK for fourth order Runge-Kutta Method

```
FURTINE2

      SUBROUTINE SHOCK(Z,T,NB,UB,UT,NI,VI,VT,C1,CC,DD,TI,H,NPTS)
*     FINDS SHOCK WAVE BEGINNING AT ANY TIME AND DEPTH TO DESIRED DEPTH
*     ASSUMES THAT IT HAS NECESSARY INITIAL AND BOUNDARY CONDITIONS
      DIMENSION UB(120),UT(120),VI(120),VT(120)
      COMMON/BLK1/DS(200),TS(200)
      REAL K1,K2,K3,K4
      K=0
      TI=T
      IF(Z.EQ.0.)GO TO 25
*       SAVE PREVIOUS SHOCK PATH IF THIS IS A CONTINUATION
      DO 30 I=1,2000
      K=I
      IF(DS(K).GT.Z)GO TO 40
 30  CONTINUE
 40   DS(K)=Z
      TS(K)=T
*     BEGIN RUNG-KUTTA TECHNIQUE
  25 D1=Z
      CALL SLOPE(TI,Z,NB,UB,UT,NI,VI,VT,C1,CC,SS,KEXC)
      K1=H*SS
       X=TI+H/2.
      Y=Z+K1/2.
      CALL SLOPE(X,Y,NB,UB,UT,NI,VI,VT,C1,CC,SS,KEXC)
      K2=H*SS
      X=TI+H/2.
      Y=Z+K2/2.
      CALL SLOPE(X,Y,NB,UB,UT,NI,VI,VT,C1,CC,SS,KEXC)
      K3=H*SS
      X=TI+H
      Y=Z+K3
      CALL SLOPE(X,Y,NB,UB,UT,NI,VI,VT,C1,CC,SS,KEXC)
      K4=H*SS
      Z=Z+(K1+2.*K2+2.*K3+K4)/6.
      TI=TI+H
      K=K+1
      TS(K)=TI
      DS(K)=Z
*     CHECKING TO SEE THAT BOUNDARY COND. HAVE NOT BEEN EXCEEDED
      IF(K.EQ.1)GO TO 50
      IF(KEXC.EQ.0)GO TO 50
      DD=DS(K)
      TI=TS(K)
      NPTS=K-1
      GO TO 100
  50 IF(Z.LT.DD)GO TO 25
      TI=((DD-D1)/(Z-D1))*(TI-(TI-H))+(TI-H)
      NPTS=K
 100 RETURN
      END
```

## Appendix C

### Function Program Input

1. Umax (UMAX) - maximum value of $u_+$ (cm/sec) for input data function (presently, the sine function).

2. $k^{1/3}/\phi_e$ (C) - snow property parameter ($cm^{2/3}$)

3. Depth (ZD) - depth (cm) of interest.

4. Shock Wave Time Step (SS) - time step (sec) used for generating the shock front.

5. Time Step at Depth (SN) - interval (sec) that v values are to be generated at depth (normally 600 sec).

6. Initial Condition Flag (I1) - parameter to indicate whether or not there are initial conditions; 1 = yes, 0 = no.

7. Umax for Initial Conditions (UIMAX) - maximum value of $u_-$ (cm/sec) for the input data function (same function as boundary conditions).

8. Output File (FN1) - name of file in which flux and time (u,t) at depth are written (may be cards, tape, etc.).

9. U Cutoff Value (CUT) - value of u (cm/sec) at depth to stop the run (-1 if run is to be stopped after 1 day).

Appendix D
RUNFC, Computer Program Simulating Input with a
Sinusoid Function

```
RUNFNC

*     PROGRAM TO GENERATE FLUX AT ANY DEPTH IN SNOWPACK
*     INPUTS ARE DEPTH,SNOW CHARACTERISTICS,AND STEP SIZES FOR ITERATION
*     BOUNDARY CONDITIONS GENERATED BY UMAX*SIN(W*T) BUT CAN BE CHANGED
*     TO ANY SIMPLE FUNCTION,IN FUNCTION FNU. AN ITERATIVE PROCEDURE IS
*     USED TO FIND U+ OR U- FOR ANY Z,T
      DIMENSION T(1000),U(1000)
      CHARACTER FN1*8
      PRINT,"BOUNDARY CONDITION PARAMETERS (CGS)"
      PRINT,"UMAX,K**(1/3)/PHI-E,DEPTH,SHOCK STEP,NOR.STEP"
      INPUT,UMAX,C,ZD,SS,SN
      ALPHA=54700.
      CS=(ALPHA**(1./3.))*C
      CC=3.*CS
*     FUNCTION TO PERSIST FOR HALF-DAY
      W=3.1415927/43200.
      PRINT,"INITIAL COND. 1=YES,0=NO"
      INPUT,I1
      IF(I1.EQ.0)GO TO 10
      PRINT,"INITIAL UMAX"
      INPUT,UIMAX
   10 PRINT,"OUTPUT FILE,U-CUTOFF(-1 IF 1 DAY RUNOUT)"
      INPUT,FN1,CUT
      OPENFILE 1,FN1,"NUMERIC"
      Z=0.
      TSTEP=0.
      TB=0.
     TIB=43200.
      UP=FNU(10.,UMAX,W)
      UM=0.
*     BEGINNING OF ROUTINE TO FIND SHOCK
*     PROJECT SHOCK FOUND AT TIME TSTEP TO NEXT TIME STEP(EULER'S)
  100 Z=Z+SS*(FNS(UP,UM,CS))
      TSTEP=TSTEP+SS
      IF(Z.GT.ZD)GO TO 200
*     FIND THE BOUNDARY CONDITION FOR TSTEP,Z
      CALL UPLUS(TB,TSTEP,Z,TF,UMAX,W,CC)
      TB=TF
      UP=FNU(TF,UMAX,W)
*     IF NO INTIAL CONDITIONS SET TO 0
      IF(I1.GT.0.)GO TO 150
      UM=0.
      GO TO 100
*     FIND THE INITIAL CONDITION FOR TSTEP,Z
  150 CALL UMINUS(TIB,TSTEP,Z,TF,UIMAX,CC,W)
      TIB=TF
      UM=FNU(TF,UIMAX,W)
      GO TO 100
*     FIND TIME OF SHOCK INTERCEPT WITH DESIRED DEPTH
  200 TI=(ZD-Z)/FNS(UP,UM,CS)+TSTEP
      TK=TI
```

RUNFNC (continued)

```
      PRINT,"SHOCK WAVE INTERSECTION,DEPTH,TIME =",ZD,TI
      CALL UPLUS(TB,TK,ZD,TF,UMAX,W,CC)
      UI=FNU(TF,UMAX,W)
*     FINDING DEPTH PROFILE UP TO SHOCK WAVE (INITIAL CONDITIONS)
      NINT=IFIX(TI/SN)+1
      TS=0.
      DO 300 I=1,NINT
      IF(I1.GT.0)GO TO 350
      U(I)=0.
      GO TO 375
 350  TIB=43200.
      CALL UMINUS(TIB,TS,ZD,TF,UIMAX,CC,W)
      U(I)=FNU(TF,UIMAX,W)
 375  T(I)=TS
      TS=TS+SN
      WRITE(1)U(I),T(I)
 300  CONTINUE
      WRITE(1)U(NINT),TI
      WRITE(1)UI,TI
*     FINDING DEPTH PROFILE AFTER SHOCK (BOUNDARY CONDITIONS)
      TS=TS+SN
      NINT=NINT+1
      T1=SN
      DO 400 I=NINT,2000
      CALL UPLUS(T1,TS,ZD,TF,UMAX,W,CC)
      U(I)=FNU(TF,UMAX,W)
      T(I)=TS
      T1=TF
      TS=TS+SN
      WRITE(1)U(I),T(I)
      L=I
      IF(CUT.LT.0..AND.TS.GT.86400)GO TO 500
      IF(U(I).LT.CUT)GO TO 500
 400  CONTINUE
 500  PRINT,"STOP TIME,U=",T(L),U(L)
      STOP
      END
      SUBROUTINE UPLUS(T1,T,Z,Y,UMAX,W,CC)
*     RETURNS BOUNDARY CONDITION TIME FOR T,Z T1 IS LOWER LIMIT OF
*     THE ITERATION INTERVAL
      T2=T
   50 Y=(T1+T2)/2.
      E=1-FNZ(FNU(Y,UMAX,W),T,Y,CC)/Z
*     .002 IS THE ITERATION CONVERGENCE CRITERIA
      IF(ABS(E).LT..002)GO TO 100
      IF(ABS(T1-T2).LT.1.E-2)GO TO 100
   60 IF(E)120,100,80
   80 T2=Y
      GO TO 50
  120 T1=Y
```

RUNFNC (continued)

```
      GO TO 50
  100 RETURN
      END
      SUBROUTINE UMINUS(T1,T,Z,Y,UIMAX,CC,W)
*     RETURNS INITIAL CONDITION TIME FOR T,Z T1&T2 ARE SEARCH INTERVAL
      T2=21600.
      TZ=T+43200.
   50 Y=(T1+T2)/2.
      X=Y-43200
      IF(ABS(X).LT.1.E-2)GO TO 100
      E=1-FNZ(FNU(Y,UIMAX,W),TZ,X,CC)/Z
      IF(ABS(T1-T2).LT.1.E-2)GO TO 100
*     .002 IS THE ITERATION CONVERGENCE CRITERIA
      IF(ABS(E).LT..002)GO TO 100
      IF(E)80,100,120
   80 T2=Y
      GO TO 50
  120 T1=Y
      GO TO 50
  100 RETURN
      END
      FUNCTION FNU(T,UMAX,W)
*     RETURNS U FOR INPUT TIME OF FUNCTION(SINE WAVE FUNCTION)
      FNU=UMAX*SIN(W*T)
      IF(FNU.LT.0.)FNU=0.
      RETURN
      END
      FUNCTION FNZ(U,T,TO,CC)
*     RETURNS Z FOR U,T  COMPUTES SLOPE OF CHARACTERISTIC
      IF(U.GT.0.)GO TO 20
      FNZ=0.
      GO TO 25
   20 FNZ=CC*(U**(2./3.))*(T-TO)
   25 RETURN
      END
      FUNCTION FNS(U1,U2,CS)
*     COMPUTES SLOPE OF SHOCK FOR GIVEN U+ AND U-
      FNS=CS*(U1**(2./3.)+U1**(1./3.)*U2**(1./3.)+U2**(2./3.))
      RETURN
      END
```