MODELLING THE GROWTH OF A PROGRADING DELTA:

Numerics, Sensitivity, Program Code and Users Guide

E.A. Calabrese and J.P.M. Syvitski

Department of Energy, Mines and Resources
Geological Survey of Canada
Sedimentary and Marine Geoscience Branch
Atlantic Geoscience Centre

Bedford Institute of Oceanography
Box 1006, DARTMOUTH, N.S.
Canada, B2Y 4A2

Geological Survey of Canada Open File #1624
Commission géologique du Canada dossier publique #1624

ABSTRACT

A linear model for the growth of a delta into a fluvially-dominated basin is outlined. The model uses three source terms to simulate (1) bedload dumping at the river mouth, (2) channelized sediment flow, and (3) hemipelagic sedimentation. A parabolic equation is used to model the downslope diffusion of previously deposited sediment. The equation is solved explicitly. Responsiveness to ten input parameters is calibrated. Results indicate delta growth is particularly sensitive to events that redistribute material after its initial deposition in the basin.

The enclosed numerical programs are in ANSI standard code and formulated for a Fortran 77 compiler. Program DELTA is the composite model that records the spatial and temporal accumulation rates within the basin, changes in the basin bathymetry, local site information such as sedimentation rates or bathymetric slope, and variations in the rate of delta front progradation. Program GRAIN models the deposition of sediment in a two-dimensional plume which spreads upon entry into a stratified basin. GRAIN computes the proportion of each size fraction deposited on the seafloor, as a function of distance from the river mouth. A users guide to operate these numerical models is provided.

DELTA MODEL

Background

The numeric model, DELTA, simulates the growth of a river delta prograding into a fluvially-dominated marine basin. The following assumptions are made about the bathymetry and dynamics of the basin.

1) Lateral variations in the position of the river channel and the shape of the delta are insignificant over the time span of the simulation.
2) Lateral cross-sections of the basin and the river mouth are rectangular.
3) The delta head is perpendicular to the inlet walls.
4) The basin has a sill.
5) The river plume is two-dimensional, maintaining a constant depth as it flows into a two-layered, stratified basin.
6) Flow velocity in the river channel is uniform.

The model is linear, distributing the total sediment deposited in any cross-section evenly across the width of the basin. Input parameters are time and tidally averaged values.

Three processes are modelled to initially deposit sediment. Much of the bedload is dumped within a few hundred metres of the delta front. The remaining bedload, in simulation of turbidity currents, is spread down-basin across may kilometers of the seafloor. Hemipelagic sediment is distributed over the entire basin. A diffusion model is next used to simulate the action of waves, local slides, tides, bioturbation and creep, in the redistribution of these primary sedimentary deposits. The model (Fig. 1) is developed and justified by Syvitski et al. (in prep.).

Bedload

In our simulation of sediment transport and deposition, we are concerned with that part of the material that actually enters the marine environment; topset deposits are not considered. We assume that a portion of bedload, α_b, may be removed by turbidity flows which bypass part of the nearshore before depositing their load as a fan in the deep part of the basin. The amount of material deposited at a distance, x, from the river mouth is a function of the slope near the river mouth and the slope of the basin floor. If the foreslope is gentle (less than some critical value, θ), we assume that no delta front failure occurs and channelized flows are not generated; bedload is dumped wholly near the river mouth. After a foreslope failure, material is deposited seaward of a point x_o, where the slope of the floor falls below θ.

To model these processes we consider the depth of the deposit per unit time integrated along the length of the basin from river mouth to sill, that is A_b, the area of bedload deposited along a longitudinal cross-setion of the basin.

(1) $$A_b = Q_b / \rho_s w_f$$

where Q_b is the bedload, ρ_s is its mass density, and w_f is the basin width. Bypass material, i.e. material deposited by turbidity currents and other related sediment gravity flows, is deposited as a linear function of x, beginning at x_o and continuing to $x_o + l_b$, where l_b is the longitudinal extent of the deposit. In the model version presented in appendix A, the deposit is twice as thick on the near-shore end as it is on the seaward end. [This type of seaward thining of turbidite units is based in part on geophysical/sedimentological data from a number of arctic fjords (Syvitski, 1984). Users should consider whether this type of thining is appropriate for their particular basin or need. Changes should be made to lines 3150 and 3200 of the subroutine BYPAS in the program GRAIN.] This approach gives bypassing, B, in units of height (i.e. thickness) per unit time.

$$(2) \qquad \begin{array}{ll} B(x,\partial h/\partial x) = 2 \propto_b A_b \,(2 - (x - x_o)/l_b) \;/\; 3\, l_b & x_o \le x \le x_b + l_b \\ B(x,\partial h/\partial x) = 0 & \text{otherwise.} \end{array}$$

We assume that turbidity flows do not carry material beyond the sill, represented by the final collection point at a distance L from the river mouth. Thus, if $x_b + l_b$ is some distance positioned beyond the sill, l_b in the above equqtion is replaced with l_b', where $l_b' = L - x_o$.

Under some conditions, the slope near the delta will be very gentle (as in Case A of Fig. 4). To avoid depositing bypass material near the river mouth on such a delta, the model chooses x_o so that water depth at this point is more than 30% of the depth of the basin.

Bedload, dumped near the river mouth, is deposited beginning at the first collection point beyond the river mouth and outward for some distance, l_d. Since the deposit is thick compared with the bypass deposits, more attention is paid to 'collection bins' adjacent to the points over which the material is dumped. Bedload is not deposited at the collection point $l_d + \Delta x$ from the river mouth. The dumping function D, in units of height (thickness) per unit time, is given by

$$(3) \qquad \begin{array}{ll} D(x) = (1 - \propto_b)\, A_b \,/\, (\,l_d - \Delta x) & \Delta x \le x \le l_d, \\ & \text{foreslope steeper than } \theta; \\ D(x) = A_b \;/\, (\,l_d - \Delta x) & \Delta x \le x \le l_d, \\ & \text{foreslope less steep than } \theta; \\ D(x) = 0 & \text{otherwise,} \end{array}$$

where Δx is the increment in x between collection points.

The longitudinal cross-section of this deposit is rhomboidally shaped. The area of sediment deposited per unit time in the first collection bin beyond the river mouth and the collection bin between l_d and $l_d + \Delta x$ is $D(x,t)\, \Delta x / 2$.

Input parameters that affect the distribution of bedload are w_f, θ, α_b, Q_b, l_b, l_d, L, and h_c, the initial height at the origin. As well as limiting the distance over which bypass deposits are distributed, L affects the slope of the initial profile, as does h_c. If the input value for $l_d < 2\,\Delta x$, the program sets the parameter at $2\,\Delta x$.

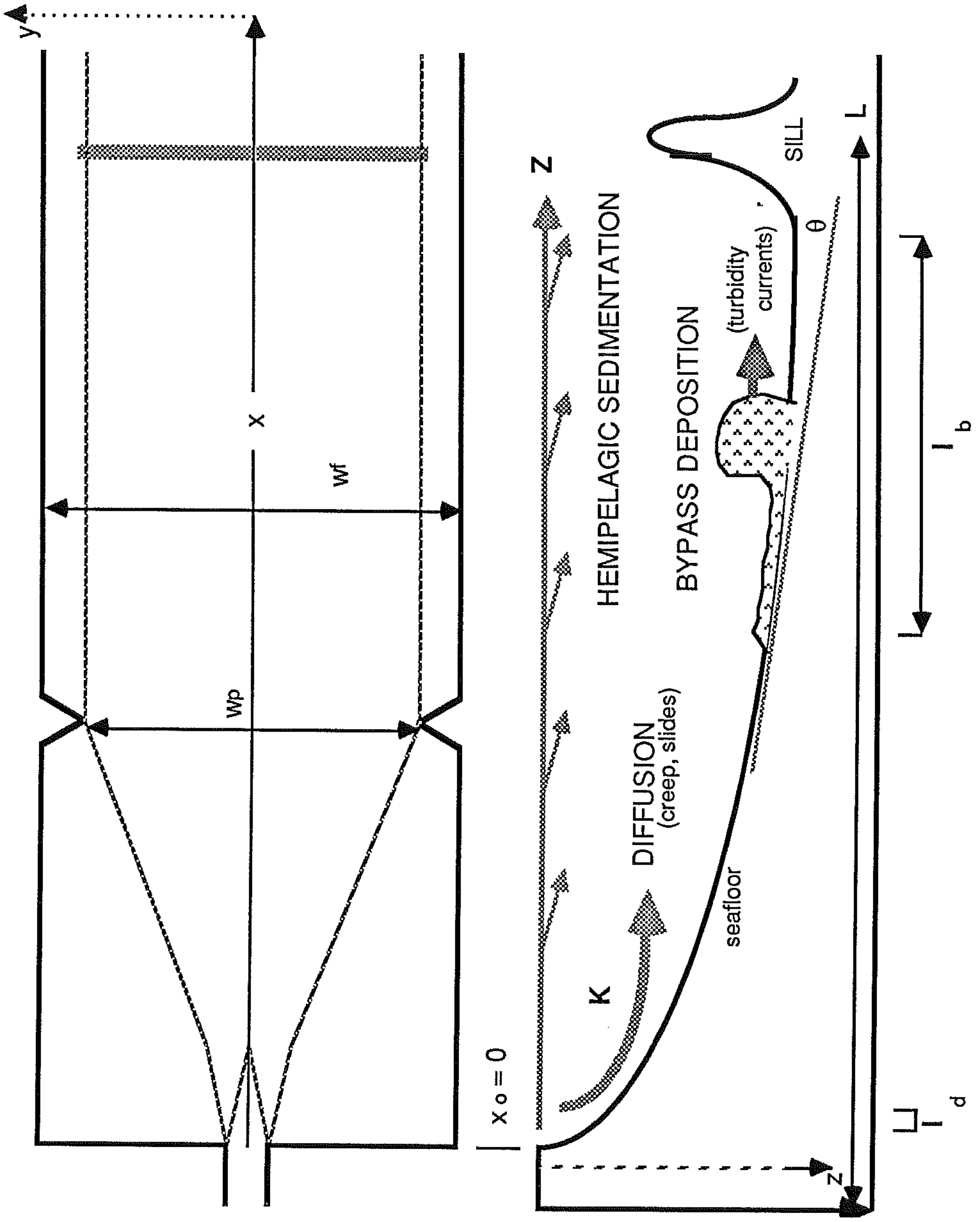

Fig. 1: Model Schematic

Hemipelagic Rain

Sedimentation rate is described as a function of grain size, initial load and time, where t_o is the time at which the sediment enters the basin. We assume that sedimentation is the only means by which particles are removed from the water column and that the removal rate is constant. From this we derive the following equation for the sedimentation rate, Z_s, in units of height (thickness) per unit time, of a single grain size.

$$(4) \qquad Z_s(x) = -\lambda_s \ \rho_s \, h_o \, C_{so} \, e^{-\lambda_s t}$$

where λ_s is the removal rate constant, which may be determined by regression analysis on SPM data, ρ_s is the mass density of the sediment, h_o is the depth of the river channel, C_{so} is the concentration of suspended sediment at the river mouth, and t is the time passed since the sediment-laden water entered the basin. The total sedimentation rate is calculated by summing values of Z_s for the individual size fractions.

Longitudinal velocity under the spreading plume is determined using a model developed by Albertson et al. (1950). They identified two regions of spread. The zone of establishment has a central core in which there has been no mixing of incoming and surrounding fluid, and hence no change from the initial velocity at the river mouth. When diffusive processes have penetrated to the centre of the plume, at x_a, the dynamics of the flow are established (Fig. 2).

Their analysis produced the following equations for u, the longitudinal component of velocity.

$$(5) \qquad u(x,y) = u_o \qquad\qquad x \le x_a,\ y \le w_c\,(1-x/x_a);$$

$$u(x,y) = u_o \exp\,[-(y + 0.5\,\pi^{0.5}\,\phi\, x - \ 0.5\, w_c) \ / \,(2 \ \phi^2\, x^2)] \qquad x \le x_a,\ y > w_c\,(1-x/x_a);$$

$$u(x,y) = u_o\,(x_a/\,x)^{0.5} \exp\,[-y^2 / \,(2 \ \phi^2\, x^2)] \qquad x > x_o;$$

where u is the longitudinal component of velocity, y is lateral distance from the centre of the plume, u_o is the velocity of the water at the river mouth, ϕ is an empirically determined constant (0.109), w_c is the width of the river mouth, and $x_a = 5.2\ w_c$.

Lateral velocity, v, under the spreading plume is determined using conservation of velocity for a two-dimensional jet (equation 6), the assumptions that the plume is symmetrical about its axis (equation 7) and that there is no lateral component of velocity at the centre of the plume.

$$(6) \qquad \partial v/\partial y \ = \ -\,\partial u/\partial x$$

$$(7) \qquad v(x,y) = -v(x,-y)$$

We consider a mesh of 2n rectangles, two rectangles wide longitudinally, centered at a distance, x_i, from the river mouth and running laterally outward from the plume axis. To obtain values for v at subsequent positions (x_i,y_j), j=1,...,n, we derive from equation 7 and the discretized form of equation 6,

$$(8) \qquad v(x_i,0) = 0$$

$$v(x_i,y_1) \approx -y_1(u(x_{i+1},0) - u(x_{i-1},0)) / 2\Delta x$$

$$v(x_i,y_j) \approx -(y_j - y_{j-1})(u(x_{i+1},y_h) - u(x_{i-1},y_h))(2\Delta x)^{-1} + v(x_i,y_{j-1}) \qquad j = 2,\dots,n$$

where $y_h = (y_j + y_{j-1}) / 2$.

Equations 5 and 8 may be used to calculate values of u and v at any position in the spreading plume.

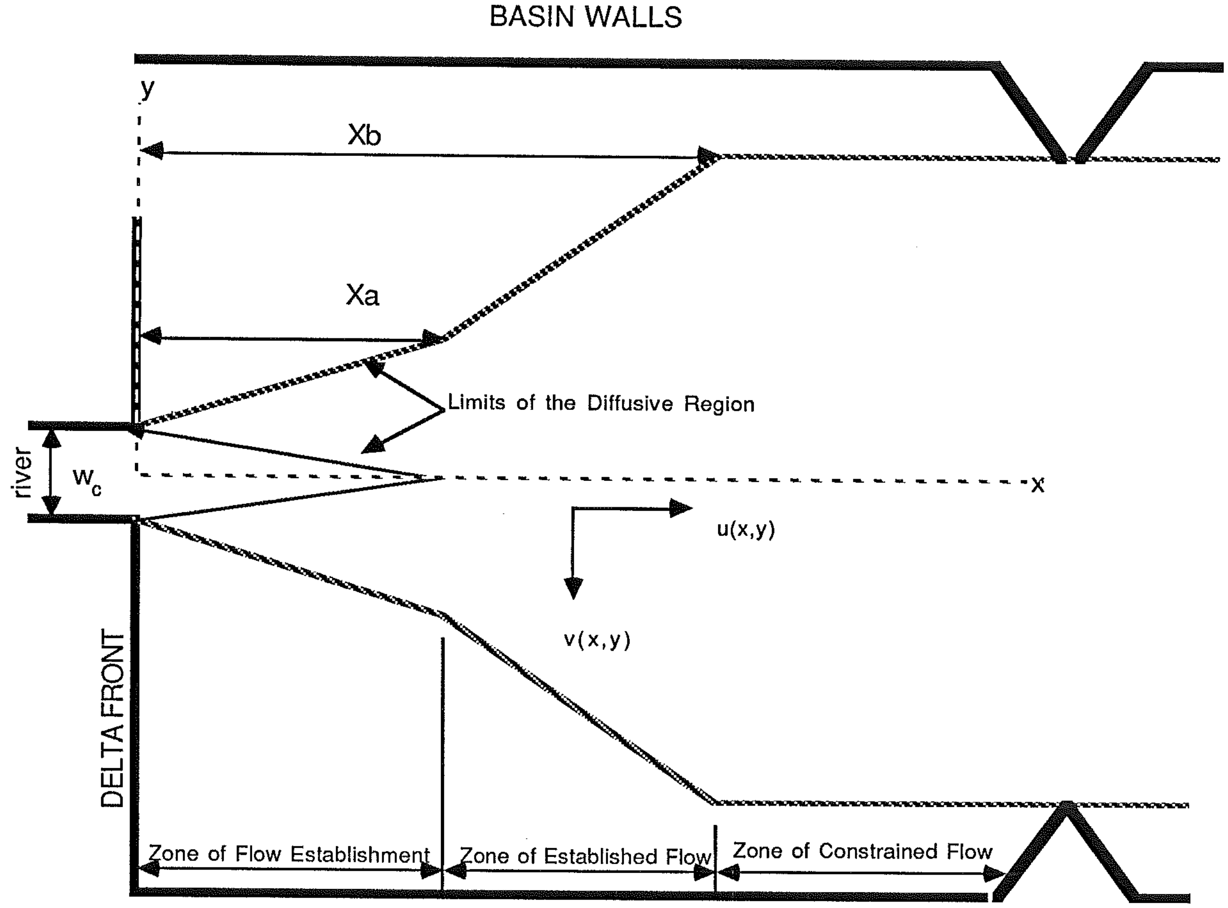

Fig. 2: Jet Diffusion

The width of the plume is constrained by the basin walls. The distance x_b at which the plume attains maximum width is obtained using the empirically determined 12° angle of spread (Albertson et al.,1950), and the final width of the plume, w_p. Beyond x_b, the residual plume velocity remains more or less constant (McClimans,1979). In the distal zone, travel time from the river mouth is given by

(9) $t(x,y,) = t(x_b,y) + (x-x_b) / u(x_b,y)$.

The program uses this velocity model to trace the pathways of columns of water as they move seaward in the basin, estimating the time required for the sediment laden parcels of water to arrive at any required point in the plume. Computed values of t are substituted into equation 4 to obtain hemipelagic sedimentation rates. Numerical integration across the plume provides the total flux deposited in any cross-section. Division of this sum by, w_f, the width of the basin, yeilds the source term Z for the linear progradation model.

Input parameters which affect the distribution of hemipelagic sediment are u_o, w_c, w_p, h_o, λ_s, ρ_s, and C_{os}. The program is designed to separate suspended sediment into four grain sizes but can be used with fewer size fractions.

Diffusion

Processes other than channelized sediment flows which move previously deposited material are assumed to operate at a rate proportional to the bathymetric slope. That is, $S = - K\, \partial h/\partial x$, where S is the rate of sediment transport and K is the coefficient of diffusion. Combining this with the conservation of mass, $\partial h/\partial t = -\partial S/\partial x$, yields the diffusion equation,

(10) $\partial h/\partial t = K\, \partial^2 h/\partial x^2$ (Culling,1960; Kenyon and Turcott,1985).

Input values which directly affect the diffusion model are K, which is estimated empirically by comparing model output with bathemetric profiles, and h_c and L, which determine the initial slope.

Numerical solution

We combine the above four processes of sediment deposition or redistribution to obtain an equation for the growth of a delta.

(11) $\partial h/\partial t = K\, \partial^2 h/\partial x^2 + B + Z + D$.

In order that there be no diffusion across the river mouth or the sill, the condition $\partial h/\partial t = 0$ is used at both boundaries. Variables are scaled by

$t^* = t/L^2$, where L is the length of the model from origin to sill, (units T / L^2)

$h^* = h/h_C$, where h_C is the initial height at the origin, (dimensionless)

$x^* = x/L$. (dimensionless)

The finite difference equation for the j+1st time step is thus

$$h^*_{i,j+1} = a(h^*_{i-1,j} + h^*_{i+1,j}) + (1-2a)\, h^*_{i,j} \;+\; b(B+D+Z) \qquad \text{(dimensionless)}$$

where $a = K\,\Delta t^* / (\Delta x^*)^2$, (dimensionless)

and $b = L^2\,\Delta t^* / h_C$. (units T / L)

The equation is solved using an explicit method and setting $\Delta t^* = 0.99\,(\Delta x^*)^2\,(2K)^{-1}$ to maintain mathematical stability.

Program Notes

DELTA sets up 1000 collection bins, the first point representing the origin, which may be the river mouth or some point landward of it, and the final point representing the sill. The number of bins actually in use during calculations depends on the the relative values of sea level and h, the height of the basin floor. If basin depth is initially set to some value less than h_c, the model will locate the furthest point from the origin that is below sea level and use it as the first collection point. As the basin fills, the number of collection bins is reduced, since the point representing the river mouth is moved toward the sill. Simulations with rising sea level, low sediment input or high rate of diffusion can cause the delta to regress and, thus, increase the number of bins. The program will abort if conditions require regressionof the seafloor beyond the origin.

The version of the model in Appendix A starts from an exponentially-shaped bathymetric profile with an upper portion of the initial height designated sediment and the remaining designated bedrock. An attempt to diffuse the bedrock will stop the simulation.

As well as producing profiles of the delta, the model generates reports detailing variations in accumulation rates with time and distance, variation in slope with time and variations in progradation rate (see Appendix C).

Program GRAIN (Appendix B), which is designed to use both the input to DELTA and its output, examines grain size fractionation as it varies with time and distance. The two programs are not completely compatible since GRAIN assumes the river mouth is initially set at the origin. Also, certain parameters may be modified during the DELTA simulation while GRAIN assumes static hydrodynamic conditions.

SENSITIVITY TESTING

Method

Sensitivy testing was conducted to calibrate the responsiveness of the simulation to variations of the model parameters, a process of particular importance in view of the averaging and estimation required to obtain input values. A base case that suggested medial conditions in a fluvially dominated basin was identified. The following values were chosen.

$l_d = 500$ m	$\theta = 1^{\circ}$
$K = 5000\ m^2 / a$	$Q_b = Q_s = 0.5\ Q_{total}$
$h_o / w_c = 0.033$	$\lambda_{coarse\ silt} = 10.6$/day
$\alpha_b = 0.5$	$\lambda_{medium\ silt} = 4.1$/day
$h_c / w_f = 0.1$	$\lambda_{fine\ silt} = 2.3$/day
$l_b = 50$ km	$\lambda_{clay} = 1.7$/day
$C_{os} = 0.25\ SC_{os}$, for all four grain sizes	

Each of the above parameters was varied while holding the others constant. Growth of the delta was simulated for 20,000 years beginning from an exponential batymetric profile, an output of a previous run that had prograded the delta 15.8 km into the basin. The distance from the origin to the sill was set at 100 km, establishing the length of the basin at 84.2 km. The depth of the basin was maintained at 438 m and the width at 4.38 km except while testing the parameter h_c/w_f.

To evaluate the sensitivity of the model, we examined the final position of the river mouth and the shape of the basin relative to the results obtained in the base run. The output profiles from each variation of the input parameters were compared the above base result using a χ^2 test on the final height at the first 401 collection points (first 40 km) beginning at the point representing the final position of river mouth. Corresponding points from the base case run were used as expected values. Before computing the χ^2 values, heights were scaled using $h^* = h / h_c$. Test of the parameter h_c / w_f is the single case where this scaling is of significance. Accumulation rates at 19,000 years were examined at 2 km intervals in the range of 1 km to 39 km from the river mouth.

Results

Variations in the lateral distance over which bypass material is spread do not modify the 20,000 year old slope nearest the river mouth (Fig. 3). Small values of l_b result in a greater overall accumulation of deposits on the proximal prodelta slopes, although they shift x_o seaward. Shortening l_b increases the progradation rate during the first 15,000 years of growth. Values of l_b greater than half the length of the basin do not significantly change the results of the run.

Reducing the distance over which bedload is dumped from 5 Δx to 2 Δx has essentially no effect on basin infilling. Mass is not conserved when l_d is set to Δx. In this case, the program makes only one deposit at the first collection point beyond the river mouth. An order of magnitude increase in l_d causes shallow slopes near the delta head but effects no other appreciable change (Fig. 4).

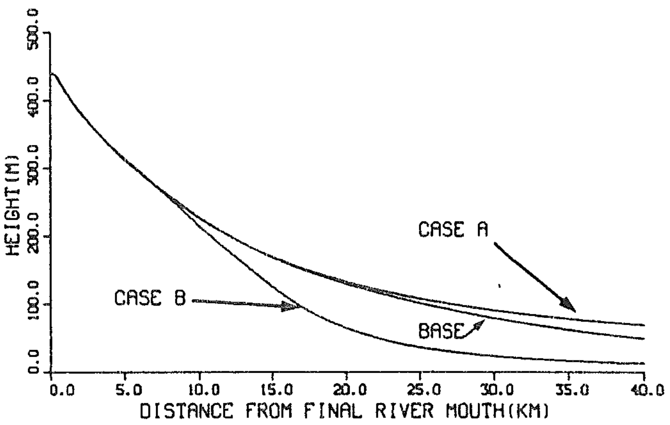

Fig. 3: Test of model sensitivity to l_b, the extent of deposits from bypassing processes. Case A: l_b = 100 km;
Case B: l_b = 10 km.

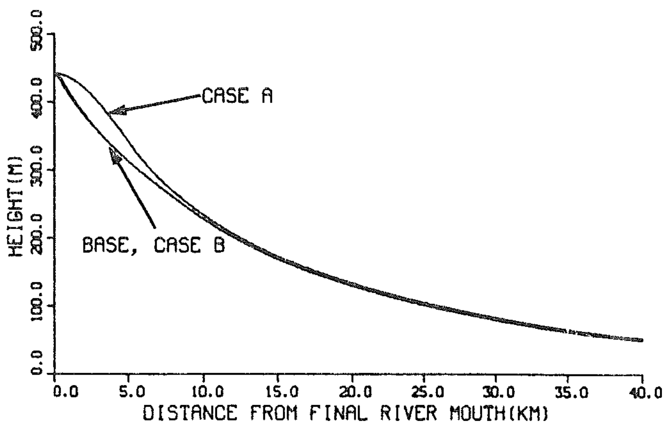

Fig. 4: Test of model sensitivity to l_d, the extent of deposits from bedload dumping at the river mouth. Case A: l_d = 5 km;
Case B: l_d = 200 m.

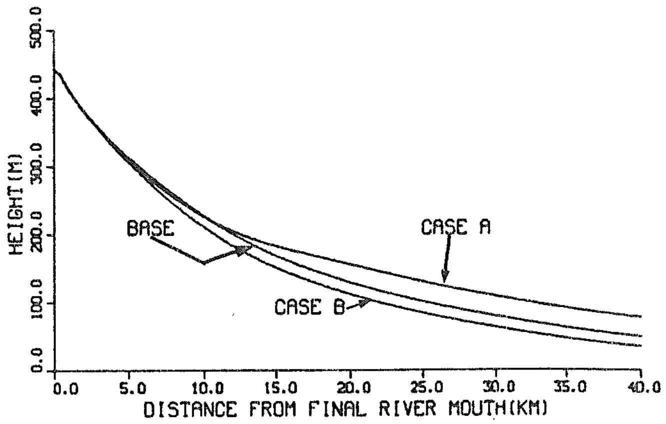

Fig. 5: Test of model sensitivity to θ, the critical slope that determines the position of bypass deposition.
Case A: θ = 0.3°; Case B: θ = 3°.

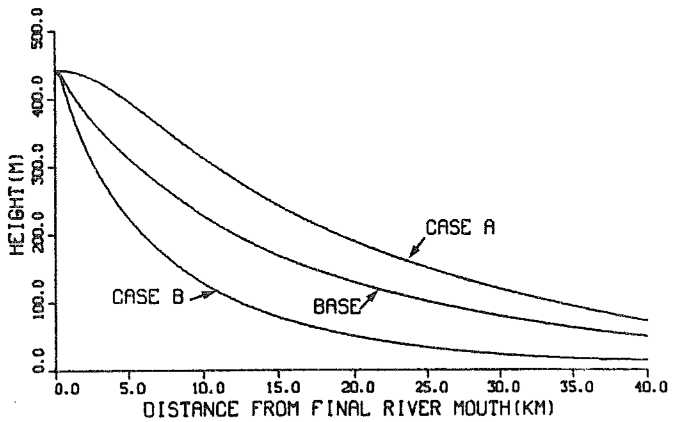

Fig. 6: Test of model sensitivity to α_b, the fraction of bedload transported by bypassing processes.
Case A: α_b = 1.0; Case B: α_b = 0.0.

The test of sensitivity for θ differed from other simulations in that the critical angle for determining whether the foreslope was steep enough for delta front failure, and thus bypassing to occur, was held at 1°, while θ was varied to check the impact of changing the slope where bypass deposition started. No check was made on water depth at x_0. This permitted deposition of bypass material at any point seaward of the river mouth. If θ is greater than 1°, bypass material is deposited within 1 km of the delta head, resulting in a greater accumulation of material in the inner basin; for low values, x_0 occurs as far out as 20 km. The model showed more sensitivity to changes in θ as θ decreases (Fig. 5).

As expected, increasing the fraction of bedload that is distributed by channelized sediment flows, results in a greater accumulation of material in the more distal parts of the basin. The change in slope from case B to case A moves x_0 6 km seaward, amplifying the seaward shift of sediment. When $\alpha_b = 1.0$, the slope near the river mouth is not steep enough to generate gravity flows during 35% of the iterations, resulting in bedload dumping at the river mouth and a moderation of model sensitivity to values of α_b greater than 0.7 (Fig. 6).

For conditions represented by our base choice, as sediment input shifts from suspended load to bedload, more material accumulates in the deeper basin and the slope near the river mouth increases sharply (Fig. 7). The impact of diffusion on accumulation rates is increased in both extreme cases. When $Q_b = Q_{total}$, $Q_s = 0$, the proximal prodelta slopes are very steep and material dumped at the river mouth is being diffused rapidly seaward. When $Q_b = 0$, $Q_s = Q_{total}$, the proximal prodelta slopes are more gentle, although less sediment accumulates in the distal parts of the basin.

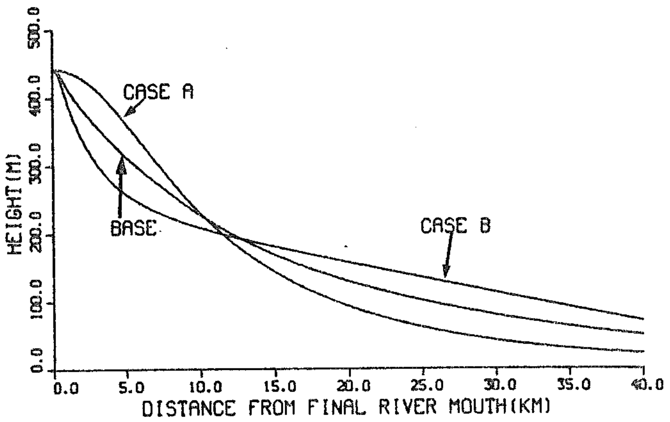

Fig. 7: Test of model sensitivity to the of suspended load to bedload
Case A: $Q_s = Q_{total}$, $Q_b = 0$
Case B: $Q_s = 0$; $Q_b = Q_{total}$.

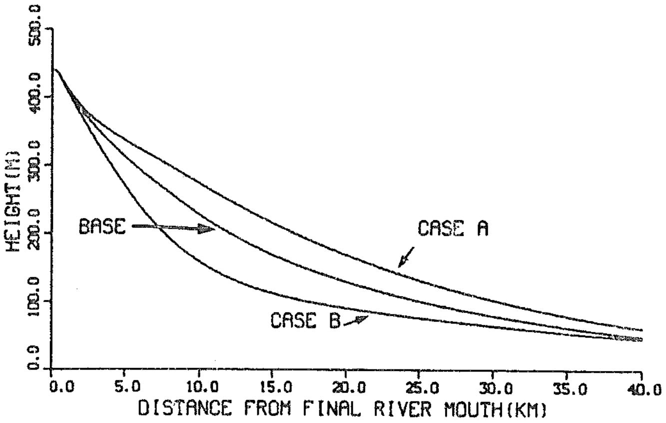

Fig. 8: Test of model sensitivity to the dis-io tribution of size fractions within the suspended load.

Case A: sediment is all clay;

Case B: sediment is all coarse silt.

Grain size in the suspended load was varied by changing C_{so} for individual size fractions with ΣC_{so} held constant. Increasing grain size of the suspended load increases the gradient in prodelta sedimentation rates and shifts the maximum deposition landward. This results in steeper slopes and more accumulation in the proximal prodelta environment (Fig. 8).

Variations of removal rate constant (Fig. 9) from the base case were as follows.

A) $\lambda_{\text{coarse silt}} = 42.7/\text{day}$, $\lambda_{\text{medium silt}} = 4.63/\text{day}$, $\lambda_{\text{fine silt}} = 0.515/\text{day}$, $\lambda_{\text{clay}} = 0.021/\text{day}$
B) $\lambda_{\text{coarse silt}} = \lambda_{\text{medium silt}} = \lambda_{\text{fine silt}} = \lambda_{\text{clay}} = 10.6/\text{day}$
C) $\lambda_{\text{coarse silt}} = \lambda_{\text{medium silt}} = \lambda_{\text{fine silt}} = \lambda_{\text{clay}} = 42.7/\text{day}$

Case A values for λ_s were calculated using Stokes settling velocity, ω, by means of the relationship $\lambda = \omega / h_o$. Cases B and C were chosen to test the effect of the assumption that settling velocity is constant for all grain sizes.

In case A, hemipelagic flux rates compared to the base case are higher in the distal regions, reflecting the greater transport distance of the finer silt and clay fractions. Large values of λ (cases B and C) cause steeper slopes and more accumulation near shore. If removal rates for fine silt and clay are very high, as in case B, a larger proportion (36%) of the suspended sediment is carried beyond the sill, contrasting with the base case (4%). The model is insensitive to a factor of two change in the removal rate (table 1).

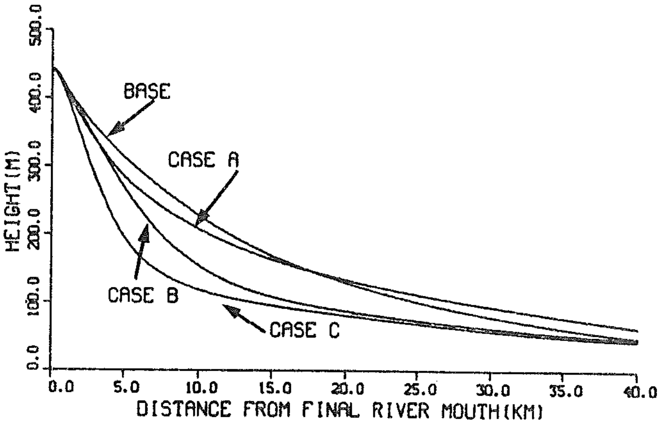

Fig. 9: Test of model sensitivity to λ, the period of removal for suspended particles from the water column.

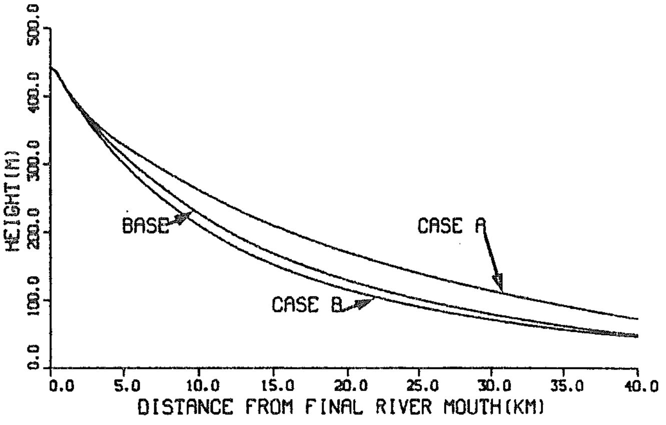

Fig. 10: Test of model sensitivity to river mouth dimensions h_o / w_c.
Case A: $h_o / w_c = 1.0$;
Case B: $h_o / w_c = 0.01$.

To test model responsiveness to the shape of the river mouth, we varied h_o / w_c while maintaining a constant cross-sectional area (Fig. 10). The removal rate constants varied by

$$\lambda_{s\,case} = \lambda_{s\,base}\, h_{o\,case} \;/\; h_{o\,base}$$

to simulate the slower removal rate of particles from a deeper water column. Here, the subscripts, case and base, indicate values used in the test run and the base run respectively. The distance from the river mouth at which hemipelagic flux reaches a maximum is similar in all cases. The gradient in the flux rate decreases as the depth of the river increases, depositing more material in the distal regions and removing greater quantities of sediment from the basin. Deltas with wide, shallow river mouths, i.e. a sandur delta, progrades faster than deltas having only one single but deep channel.

Sensitivity to changes in basin geometry was tested by varying h_c/w_f while maintaining the cross-sectional area of the basin. Except in the range, $0.019 < h_c/w_f < 0.023$, the model is relatively insensitive to changes in this parameter. For values of h_c/w_f greater than 0.023, the foreslope remains steep enough for the model to simulate sediment gravity flows during all iterations. For parameter values of 0.019 or less, no channelized flows are generated during the run. Increased bedload dumping at the river mouth correlates with a slower overall delta front progradation rate and a sharp increase in deviation from the base profile (Fig. 11 and table 1).

A large coefficient of diffusion, representing a high rate of sediment transport along the prodelta slopes, reduces accumulation near shore and increases deposition in the intermediate and distal regions of the basin (Fig. 12). The shallow slopes that result from large K decrease the frequency of channelized flows. For K=25,000 m^2/a, bedload was dumped wholly at the river mouth during 96% of the iterations.

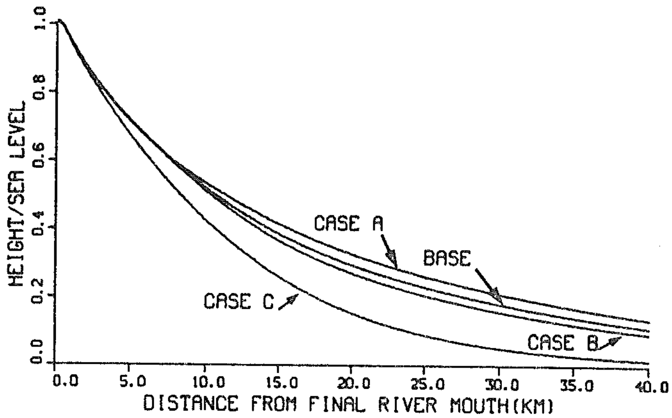

Fig. 11: Test of model sensitivity to basin geometry, h_c/w_f.
Case A: $h_c/w_f = 0.2$; Case B: $h_c/w_f = 0.05$;
Case C: $h_c/w_f = 0.01$

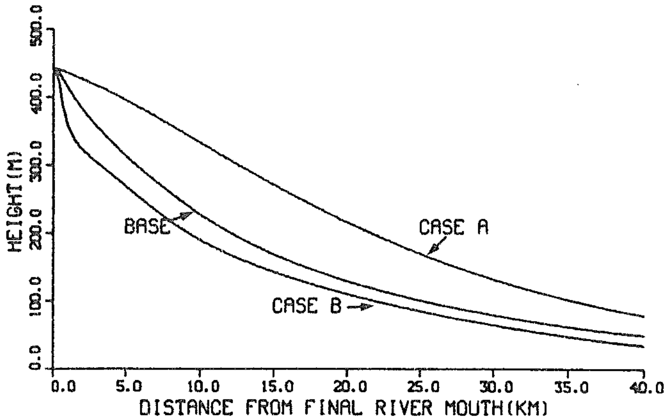

Fig. 12: Test of model sensitivity to the coefficient of diffusion, K.
Case A: K = 25,000 m^2/a;
Case B: K = 1,000 m^2/a.

Table 1: Relative progradation of the river mouth and χ^2 values for sensitivity tests. The third column gives the position of the final river mouth relative to the base case in km. χ^2 values are calculated using the first 401 points from the river mouth. Base case results are used as expected values.

Parameter	**Value**	**Relative Progradation**	χ^2
l_b	10 km	4.1	19.72
	100 km	-1.0	1.14
l_d	200 m	0	0.00
	5 km	-0.9	0.56
θ	0.3°	-2.3	5.42
	3.0°	1.6	2.04
$C_{course\ silt}$, C_{clay}	C_{total}, 0	0.8	7.74
	0, C_{total}	-0.9	6.63
h_o / w_c	0.01	1.4	0.86
	1.0	-7.5	8.93
h_c / w_f	0.013	-4.4	22.00
	0.019	-4.4	21.30
	0.023	1.2	1.13
	0.2	-1.3	1.18
K	1000 m^2/a	2.7	3.84
	25000 m^2/a	-17.2	34.95
α_b	0.0	6.9	34.08
	1.0	-12.7	20.58
Q_b / Q_t , Q_s / Q_t	1,0	-2.8	7.12
	0,1	3.4	9.23
λ	Case A *	-7.7	1.69
	Case B	3.5	8.89
	Case C	4.5	18.48
	Case D	0.0	0.00
	Case E	0.0	0.00

* Case A: $\lambda_{coarse\ silt} = 42.7/day$, $\lambda_{medium\ silt} = 4.63/day$, $\lambda_{fine\ silt} = 0.515/day$, $\lambda_{clay} = 0.021/day$

Case B: $\lambda_{coarse\ silt} = \lambda_{medium\ silt} = \lambda_{fine\ silt} = \lambda_{clay} = 10.6/day$

Case C: $\lambda_{coarse\ silt} = \lambda_{medium\ silt} = \lambda_{fine\ silt} = \lambda_{clay} = 42.7/day$

Case D: $\lambda_s = 2.0\ \lambda_{s\ base}$ for each grain size

Case E: $\lambda_s = 0.5\ \lambda_{s\ base}$ for each grain size

Conclusions

The growth of a delta is most sensitive to those processes which transfer marine deposits from the nearshore into deeper water. The parameters which effect the greatest change in simulation output are

1) the rate of seaward transport of accumulated sediment by diffusive processes such as creep and slides,
2) the fraction of bedload eventually redistributed from the foreslope by sediment gravity flows,
3) the longitudinal extent of bypass deposits made in the deep basin where this distance is less than half the basin length,
4) changes in bathymetry that affect the frequency of channelized sediment flows.

The model showed moderately high sensitivity to order of magnitude changes in the removal rate of sediment from the river plume. Basin infilling responds moderately to variations in

1) the distribution of sediment between suspended load and bedload,
2) grain size fractionation within suspended load,
3) bathymetry of the river mouth,
4) the slope, θ, of the basin floor where sediment flows begin to redeposit material, if $\theta < 0.5^{o}$.

Delta growth is moderately insensitive to changes in

1) the longitudinal extent of bypass deposition, where this distance is greater than half the basin length,
2) the longitudinal extent of bedload dumping near the river mouth,
3) critical slope, θ, if $0.5^{o} < \theta < 1.5^{o}$.

The model is highly insensitive to

1) moderate changes in the removal rate of sediment from the river plume,
2) variations in θ, for $\theta > 1.5^{o}$.

Actual height of accumulated sediment varied greatly with changes in basin bathymetry, although relative heights (h / h_c) and progradation rate were moderately insensitive except when variations in parameters affected transport of bedload.

ACKNOWLEDGEMENTS

The authors would like to thank the Geological Survey of Canada for their support in this project. Dr. B. Boudreau was instrumental in the development of an earlier version of our DELTA program. We are grateful to Drs. C. Amos and D. Forbes for their critical comments on this open file.

REFERENCES

Albertson, M.L.; Dai, Y.B.; Jensen, R.A.; and Hunter, R.; 1950, Diffusion of submerged jets. Amer. Soc. Civil Eng. Trans., v. 115, p. 639-697.

Culling, W.E.H.; Analytical theory of erosion. J. Geology, 68,336-344, 1960.

Kenyon, P.M. and Turcotte, D.L.; Morphology of a delta prograding by bulk sediment transport. Geol. Soc. Am. Bull., 96, 1457-1465, 1985.

McClimans, T.A.; 1979, On the energetics of river plume entrainment. Geophys. Astrophys. Fluid Dynamics, v.13, p. 67-81.

Syvitski, J.P.M.; Smith, J.N.; Boudreau, B.; and Calabrese, E.A.; submitted, Basin sedimentation and the growth of prograding deltas. J. Geophys. Res.

Syvitski, J.P.M. (Compiler) 1984. Sedimentology of Arctic Fjords Experiment: HU83-028 data report, Vol.2. Can. Data Rep. Hydrogr. Ocean Sci. 28: 1130 p.

Notation

A_b	area of bedload deposited in a longitudinal cross-section
B	rate of deposition of bypass deposits in units of height per time
C_o	concentration of suspended sediment at the river mouth
D	rate of deposition of bedload dumped at the river mouth in units of height per time
h	height of basin floor
h_c	characteristic height, height at model origin
h_o	depth of the river channel
l_b	extent of deposits from bypassing processes
l_d	extent of deposits from bedload dumping at the river mouth
K	coefficient of diffusion
L	distance from origin to sill
Q_b	bedload entering marine environment
Q_s	suspended load
Q_{total}	total load
S	rate of transport of material by diffusive mechanisms
t	time
u	longitudinal component of river plume velocity
v	lateral component of river plume velocity
w_c	width of river channel
w_f	basin width
w_p	maximum width of river plume
x	longitudinal distance
x_a	distance from river mouth where flow dynamics are established
x_b	distance from river mouth where plume attains maximum width
Δx	increment in x for the numerical solution, width of a collection bin
x_o	position beyond which sediment is deposited from bypassing processes
y	lateral distance from river plume axis
Z	hemipelagic sedimentation rate in units of height per time
α_b	fraction of bedload transported by bypassing processes
λ	period of sediment removal from water column
ω	Stokes settling velocity
ρ	mass density
θ	critical slope that determines position of bypass deposits

Subscripts, Superscripts

s	of a particular grain size
base	value in the base case
*	scaled variable

```
      PROGRAM DELTA
*     CODE IS ANSI STANDARD
******************************************************************************
* MAINLINE

*      BACKGROUND
*               JANUARY, 1987
*               DELTA IS A LINEAR MODEL FOR THE GROWTH AND
*               PROGRADATION OF A DELTA IN A RECTANGULAR BASIN
*               THE MODEL USES THREE SOURCE TERMS:
*                 A) BYPASSING(X,T,DH/DT)
*                 B) DUMPING(X,T)
*                 C) HEMIPELAGIC FLUX(X,T)
*               THE DEPOSITED SEDIMENT IS SMEARED DOWNSLOPE
*               USING A DIFFUSION EQUATION (THE SORT USED IN A HEAT
*               DIFFUSION MODEL)
*               THE NUMERICAL SOLUTION IS EXPLICIT.
*
*               DETAILED BACKGROUD AND ALGORITHM DOCUMENTATION
*               IS AVAILABLE ELSEWHERE.
*
*********************************************************************
*
*    AUTHOR: E.A. CALABRESE
*            ATLANTIC GEOSCIENCE CENTRE
*            GEOLOGICAL SURVEY OF CANADA
*            BOX 1006, DARTHMOUTH, N.S.
*            CAN.   B2Y 4A2

*    MODEL DEVELOPED BY
*          J.P.M. SYVITSKI, GEOLOGICAL SURVEY OF CANADA,
*             BEDFORD INSTITUTE OF OCEANOGRAPHY,
*             BOX 1006, DARTHMOUTH, N.S.  B2Y 4A2
*          J.N. SMITH, ATLANTIC OCEANOGRAPHIC LABORATORY,
*             BEDFORD INSTITUTE OF OCEANOGRAPHY,
*             BOX 1006, DARTHMOUTH, N.S.  B2Y 4A2
*          B. BOUDREAU, DEPARTMENT OF OCEANOGRAPHY
*             UNIVERSITY OF BRITISH COLUMBIA, VANCOUVER, B.C.  V6T 1W5
*          AND AUTHOR
*********************************************************************
*
*     VARIABLES
*

*   A -- COEFFICIENT IN FINITE DIFFERENCE EQUATION.
*        DIFSN * TSTEP / (XINC**2)

*   B -- COEFFICIENT IN FINITE DIFFERENCE EQUATION
*        1 - 2 * DIFSN * TSTEP / (XINC**2)
*   BINACC -- (BINS ACROSS) NUMBER OF BINS IN A CROSS-SECTION OF
*             RIVER PLUME PERPENDICULAR TO FJORD WALLS. USED IN
*             CALCULATING SUSPENDED SEDIMENT DEPOSITION RATE(FLXTRM)
*   BINS -- NUMBER OF COLLECTION BINS ALONG THE LENGTH OF THE
*           DELTA (PTS-1)
*   BORDER -- VALUE OF Y (DISTANCE FROM PLUME CENTRE) WHERE
*             PLUG FLOW BORDERS DIFFUSIVE REGION IN ZONE OF ESTABLISHMENT.
*             USED IN MODELLING SUSPENDED SEDIMENT DEPOSITION.
*   BYPASS -- VECTOR FOR THE BYPASS SOURCE TERM.
*             STORES THE DEPTH OF BYPASS MATERIAL TO BE
*             DEPOSITED AT EACH POINT ALONG THE PROFILE.

*   C -- EXPERIMENTALLY DETERMINED CONSTANT FROM VELOCITY FIELD
*        MODEL (0.109). USED IN CALCULATING HEMIPELAGIC SOURCE TERM.
*   CHANNL -- (CHANNEL WIDTH) WIDTH OF THE RIVER MOUTH IN METRES.
*   CNTN -- CONCENTRATION. A VECTOR CONTAINING THE CONCENTRATION
*           OF SUSPENDED SEDIMENT AT THE RIVER MOUTH. BROKEN DOWN BY
*           GRAIN SIZE. READ AS G/M**3, CONVERTED TO KG/M**3
*   CNTRL -- (CONTROL) THE VARIABLE THE USER SPECIFIES TO CONTROL
*            THE OUTPUT OF PROFILES AND THE LENGTH OF THE RUN.
*            T = TIME(IE, RUN FOR 10000 YEARS,PLOT EVERY 1000 YEARS)
```

```
*            D = DISTANCE PROGRADED(IE. PROGRADE 30 KM,PLOT EVERY
*                5 KM.)

*    D -- COEFFICIENT IN FINITE DIFFERENCE EQUATION
*         (LENGTH**2)*TSTEP /UNTHGT
*    DAY2SC -- NUMBER OF SECONDS IN A DAY
*    DEGREE -- SLOPE OF PROFILE WHERE BYPASS STARTS(DEGREES)
*              ALSO USED TO DETERMINE IF FORESLOPE IS STEEP ENOUGH FOR
*               BYPASSING TO OCCUR.
*    DENSTY -- VECTOR CONTAINING MASS DENSITY OF EACH OF THE GRAIN
*             SIZES
*    DEPTH-- DEPTH OF RIVER CHANNEL AT MOUTH
*    DETIME -- (DETAIL TIME) TIME AT WHICH DETAILS ABOUT SEDIMENTATION
*              RATE ARE TO BE RECORDED. INPUTTED AS NUMBER OF YEARS,
*              IMMEDIATELY SCALED.
*    DG2RAD -- FACTOR TO CONVERT DEGREES TO RADIANS(PI/180)
*    DIFSN -- COEFFICIENT OF DIFFUSION (M**2/YR)
*    DMPALL -- DEPTH OF SEDIMENT DUMPED IF THE SLOPE IS TOO
*              SHALLOW TO PERMIT BYPASSING
*    DMPLGT --  DISTANCE OVER WHICH TO DEPOSIT BEDLOAD TO BE DUMPED
*               SEE DISCUSSION OF LINE 8 INPUT IN APPENDIX C
*    DOFTEN --  THE NUMBER OF PROGRADATIONS BETWEEN PLOTS. USED IF
*               CONTROL IS BY DISTANCE PROGRADED
*    DRECTN -- (DIRECTION) EITHER F=FORWARD TO INDICATE PROGRADATION
*              OR B=BACKWARD TO INDICATE REGRESSION. USED IN WRITING
*              TO OR READING FROM FILE ADVANCE, WHICH IS A LIST OF
*              TIMES THE MODEL IS MOVED.
*    DSTINC -- UNSCALED DISTANCE BETWEEN COLLECTION POINTS
*              (METRES)

*    E -- 2.71828182846...
*    ENDCNT -- NUMBER OF ITERATIONS BETWEEN STORAGE OF HEIGHT AND
*              CALCULATON OF CHANGE IN HEIGHT. USED IN CALCULATING
*              TOTAL ACCUMULATION OF SEDIMENT
*    ENDDMP --  (END DUMP) INDEX OF THE LAST POINT WHERE BEDLOAD IS
*               DUMPED
*    ENDRUN -- INDEX OF THE LAST POINT WHERE BYPASS IS DEPOSITED

*    FLOOR --  VECTOR THAT REPRESENTS THE HEIGHT OF THE BEDROCK. TAKEN
*              TO BE INTSED*UNTHGT METRES BELOW INITIAL PROFILE.
*    FLXTRM -- VECTOR FOR HEMIPELAGIC*SOURCE TERM.
*              STORES THE SCALED AVERAGE SUSPENDED SEDIMENT
*              DEPOSITED PER ITERATION AT EACH COLLECTION POINT ALONG THE
*              PROFILE.

*    GRAINS -- NUMBER OF GRAIN SIZE FRACTIONS BEING CONSIDERED

*    H -- VECTOR WHICH STORES THE SCALED CURRENT HEIGHT OF THE
*         PROFILE AT THE COLLECTION POINTS.
*    HALFUP -- IN THE GRID USED FOR CALCULATING HEMIPELAGIC RAIN-
*              THE Y-COORDINATE OF THE POINT HALF A SQUARE OUTWARD
*              FROM CURRENT POINT
*    HEAVY -- DENSITY OF COARSE SAND
*    HGTCHG -- HEIGHT CHANGE - USED IN CALCULATING SLOPE
*    HLAST -- (LAST HEIGHT) TEMPORARY STORAGE FOR HEIGHT DURING ITERATION.
*    HTEMP -- (TEMP HEIGHT) TEMPORARY STORAGE FOR HEIGHT DURING ITERATION.

*    I -- USED AS AN INDEX
*    INTSED -- (INITIAL SEDIMENT) SCALED DEPTH OF THIN LAYER OF INITIAL SEDIMENT

*    K -- COEFFICIENT OF X IN POWER OF E,USED TO SHAPE THE EXPONENTIAL
*         INITIAL PROFILE OF THE BASIN
*    K -- ALSO USED AS POWER OF E IN CALCULATING THE X-COMPONENT OF
*         VELOCITY.

*    LMDA --   (LAMDA) REMOVAL RATE CONSTANT FOR HEMIPELAGIC RAIN
*             READ IN UNITS 1/DAY. CONVERTED IMMEDIATELY TO
*             1/SEC
*    LENGTH -- DISTANCE FROM MODEL ORIGIN TO SILL. READ IN IN KM.
*              CONVERTED IMMEDIATEDLY TO METRES.
*    LEVEL -- STATEMENT FUNCTION,CHANGE IN SEALVL AS A FUNCTION OF TIME
```

```
*           WHERE TIME IS IN YEARS.
*   LOWPT — INDEX OF COLLECTION POINT ON TOPSET WITH LOWEST HEIGHT.
*   LXAXIS — LENGTH OF THE PLOT'S X-AXIS IN INCHES
*   LYAXIS — LENGTH OF THE PLOT'S Y-AXIS IN INCHES

*   M — M = L * I0  = L * C0 * DEPTH / DENSITY
*        COEFFICIENT OF E IN EQUATION  Z = L*C0*DEPTH/DENSITY * (E**(-L*T))
*        Z IS THE RATE OF DEPOSITION OF SUSPENDED SEDIMENT, L IS
*        THE REMOVAL RATE CONSTANT, C0 IS THE INITIAL CONCENTRATION,
*        OF SEDIMENT IN THE WATER COLUMN, AND T IS TIME.
*   MSGANG — ANGLE AT WHICH THE TIME OF THE PROFILE MESSAGE IS TO
*             BE PRINTED

*   N — N = -L * T
*        POWER OF E IN EQUATION  Z = L*I0* (E**(-L*T)) WHERE
*        Z IS THE RATE OF DEPOSITION OF SUSPENDED SEDIMENT, L IS
*        THE REMOVAL RATE CONSTANT, I0 IS THE INITIAL INVENTORY
*        OF SEDIMENT IN THE WATER COLUMN, AND T IS TIME.
*   NEWTIM — (NEW TIME) TIME OF PROGRADATION NEWLY READ FROM ADVANCE
*   NUMPTS — NUMBER OF POINTS FOR WHICH THE DETAILS ABOUT ACCUMULATION
*             ARE REQUIRED
*   NXTPRT — TIME NEXT PROFILE IS REQUIRED (OR DISTANCE FROM ORIGIN IF
*             CONTROL BY DISTANCE)
*   NXTTIM — (NEXT TIME) THE NEXT TIME(IN YEARS) AT WHICH TO
*             CALCULATE PROGRADATION RATE

*   OFTEN — INCREMENT IN INDEX FOR PRINTING ACCUMULATION DETAILS.
*            (IE, OUTPUT DETAILS FOR EVERY OFTEN POINTS)
*   OLDTIM — (OLD TIME) THE LAST TIME READ FROM ADVANCE
*   OUTSED — HEMIPELAGIC FLUX DEPOSITED(CM/A)
*   OUTSLP — SLOPE IN DEGREES OF AT THE POINT WHERE SLOPE IS BEING
*             TRACKED AT INTERVALS.
*   OUTSRC — SOURCE DEPOSITED( HEMIPELAGIC + BYPASS) IN CM/A
*   OUTTIM — (OUTPUT TIME) THE TIME(IN YEARS) AT WHICH TRACKING
*             INFORMATION IS PRINTED
*   OUTTOT — TOTAL CHANGE IN SEDIMENT DEPTH(CM/A)

*   P — COEFFICIENT IN CALCULATING BYPASS DEPOSITED
*        CHOSEN SO BYPASS DECREASES LINEARLY WITH DEPOSITION AT FIRST POINT
*        OF BYPASS TWICE AS THICK AS DEPOSITION AT LAST POINT
*   PI — CIRCUMFERENCE/DIAMETER
*   PLUME — FINAL WIDTH OF RIVER PLUME(WHEN NO LONGER SPREADING)
*   PNTFLX —  VECTOR OF COLLECTION POINTS IN A CROSS-SECTION OF
*            THE DELTA PERPENDICULAR TO FJORD WALLS. USED IN CALCULATING
*            SUSPENDED SEDIMENT DEPOSITION RATE
*
*   PORQB — THAT FRACTION OF BOTTOM-LOAD THAT IS TO BE DEPOSITED
*            AS BYPASS
*   POSTN — DISTANCE DOWN FJORD FROM THE POINT WHERE WE BEGAN DEPOSITING
*            BYPASS MATERIAL(METRES)
*   PRINT — TRUE IF PROFILE REQUIRED DURING CURRENT ITERATION
*   PROFLS —  NUMBER OF PROFILES TO BE PRODUCED WITH CURRENT SET
*              OF PARAMETERS
*   PRTCNT — NUMBER OF PROFILES PRINTED FOR CURRENT SET OF PARMS
*   PRTDST — IF CONTROL IS BY DISTANCE PROGRADED, THE DISTANCE TO
*             PROGRADE BETWEEN PLOTS OF PROFILES.
*   PTS — NUMBER OF POINTS AT WHICH SEDIMENTATION IS BEING CALCULATED
*   PTSACC — THE NUMBER OF COLLECTION POINTS IN A CROSS-SECTION
*           PERPENDICULAR TO FJORD WALLS. USED FOR COMPUTING
*           HEMIPELAGIC FLUX
*   PTSLP — SLOPE IN DEGREES AT THE POINT WHERE ACCUMULATION IS BEING
*            RECORDED AT TIME INTERVALS.

*   QB — BOTTOM LOAD ENTERING FJORD, BYPASS + DUMP (KG/S)
*   QBTHIN — AREA OF BOTTOM LOAD DEPOSITED IN A PROFILE(SQ. M/YR).
*            IE, VOLUME OF BOTTOM-LOAD DEPOSITED / WIDTH OF FJORD
*            FLOOR
*   QBYPAS — THE AREA OF BEDLOAD TO BE DEPOSITED AS BYPASS IN A
*           PROFILE. THIS IS A SCALED VALUE.
*   QDOWN — DEPTH OF BEDLOAD DUMPED AT DELTA HEAD. MAY BE
*            QDUMP OR DMPALL
```

```
*    QDUMP -- SCALED DEPTH OF BEDLOAD TO DUMP ON EACH ITERATION
*             GIVEN THAT BYPASS IS DEPOSITED SEPARATELY
*    QUIT -- TRUE IF AN ATTEMPT IS MADE TO DIFFUSE THE FLOOR
*            OR TO PROGRADE OR REGRESS THE MODEL BEYOND THE BOUNDARIES
*            OF THE SYSTEM
*    QTDST -- (QUIT DISTANCE)TOTAL DISTANCE TO PROGRADE MODEL FOR
*             CURRENT SET OF PARAMETERS

*    RATE -- PROGRADATION RATE
*    RUNNUM -- NUMBER OF POINTS OVER WHICH TO DEPOSIT BYPASSING
*              MATERIAL
*    RUNOUT -- LENGTH OF FJORD OVER WHICH TO DEPOSIT BYPASSING
*              MATERIAL

*    SEAINT -- INITIAL DEPTH OF BASIN
*    SEALVL -- HEIGHT OF WATER ABOVE H=0
*    SHALOW --  TRUE IF FORESLOPE IS TOO SHALLOW TO PERMIT BYPASSING.
*    SLOPEX -- PERIODICALLY, THE SLOPE OF THE POINT A GIVEN DISTANCE
*              FROM CURRENT RIVER MOUTH IS RECORDED. THIS DISTANCE IS
*              SLOPEX. READ IN UNITS KM. CONVERTED TO METRES.
*    SLPPT -- INDEX OF THE COLLECTION POINT CORRESPONDING TO SLOPEX
*    SPREAD -- LONGITUDINAL DISTANCE OVER WHICH BYPASS IS SPREAD
*              (METRES)
*    START -- INDEX OF THE POINT THAT REPRESENTS THE RIVER MOUTH

*    T -- (TIME) A VECTOR USED FOR ACCUMULATING THE TIME REQUIRED
*         FOR EACH COLUMN OF WATER TO REACH THE CURRENT DISTANCE
*         FROM THE FJORD HEAD. USED TO CALCULATE FLUX
*    TANMAR -- THE DIFFERENCE BETWEEN THE SCALED HEIGHT AT TWO
*              ADJACENT COLLECTION POINTS THAT REPRESENTS THE
*              SLOPE WHERE BYPASSING BEGINS
*    TANSLP -- (TANGENT) TAN OF SLOPE
*    TIMCOR -- INVERSE OF SCALING FACTOR FOR TIME(TIME CORRECTION)
*    TIMCTL -- TRUE IF THE OUTPUT OF PROFILES AND THE LENGTH
*              OF THE THE RUN IS TO BE CONTROLLED BY TIME PASSED.
*              FALSE IF THE MODEL IS TO BE CONTROLLED BY DISTANCE
*               PROGRADED
*    TIME -- SCALED TIME PASSED SINCE BEGINNING OF RUN
*    TINC -- TIME REQUIRED FOR A COLUMN OF WATER TO ADVANCE ACROSS
*            ONE SQUARE IN THE GRID USED FOR CALCULATING HEMIPELAGIC
*            FLUX
*    TOFT -- TIME(IN YEARS) BETWEEN CALCULATION OF PROGRADATION RATES
*    TOFTEN -- SCALED TIME INTERVAL BETWEEN PROFILE OUTPUTS
*    TOPX -- LARGEST VALUE OF X(DSTANCE DOWN FJORD) ON PLOT(LENGTH)
*    TOPY -- LARGEST VALUE OF Y(HEIGHT) ON PLOT
*    TRACKX -- PERIODICALLY THE SLOPE AND ACCUMULATION RATES
*              A GIVEN DISTANCE FROM THE ORIGIN ARE RECORDED.
*              THIS DISTANCE IS TRACKX.READ AS KM, CONVERTED TO METRES.
*    TRKPT -- INDEX OF THE COLLECTION POINT THAT REPRESENTS TRACKX
*    TSTEP -- SCALED TIME BETWEEN ITERATIONS. CHOSEN TO JUST MAINTAIN STABILITY

*    UNTHGT -- UNIT HEIGHT(METRES). FOR SCALING HEIGHT.
*              INITIAL HEIGHT AT THE ORIGIN

*    V0 -- VELOCITY OF RIVER WATER AT RIVER MOUTH(M/SEC)
*    VX -- VECTOR, X-COMPONENTS OF VELOCITY AT THE COLLECTION PTS
*          IN THE CROSS-SECTION PERPENDICLAR TO FJORD WALLS.
*          USED IN COMPUTING HEMIPELAGIC FLUX
*    VXBACK -- THE X-COMPONENT OF VELOCITY ONE SQUARE CLOSER TO THE
*              RIVER MOUTH AND HALF A SQUARE FARTHER FROM THE CENTRE
*              OF THE PLUME THAN THE CURRENT POINT.
*    VXFRD -- THE X-COMPONENT OF VELOCITY ONE SQUARE FARTHER FROM THE
*             RIVER MOUTH AND HALF A SQUARE FARTHER FROM THE CENTRE
*             OF THE PLUME THAN THE CURRENT POINT.
*    VY -- Y-COMPONENT(LATERAL COMPONENT) OF VELOCITY IN THE RIVER PLUME
*    VYINC -- INCREASE IN THE Y-COMPONENT OF VELOCITY AS WE MOVE OUTWARD
*             FROM THE CENTRE OF THE RIVER PLUME ONE SQUARE FROM LAST POINT.


*    WIDTH -- WIDTH OF CHANNEL AT RIVER MOUTH
```

```
*   X -- VECTOR CONTAINING THE DISTANCE OF EACH OF THE COLLECTION
*        POINTS FROM THE RIVER MOUTH. USED IN PLOTTING
*   X0 -- THE DISTANCE FROM THE RIVER MOUTH WHERE PLUME DYNAMICS
*         CHANGE FROM ZONE OF ESTABLISHMENT TO ZONE OF ESTABLISHED
*         FLOW
*   XB -- DISTANCE FROM RIVER MOUTH WHERE PLUME ACHIEVES
*         MAXIMUM WIDTH
*   XBACK -- X-COORDINATE OF THE POINT IN THE GRID ONE SQUARE CLOSER
*            TO THE RIVER MOUTH THAN CURRENT POINT
*   XBPNT -- THE INDEX OF THE COLLECTION POINT THAT REPRESENTS THE
*            THE POINT IN THE DELTA WHERE THE PLUME REACHES MAXIMUM
*            WIDTH
*   XFRD -- X-COORDINATE OF THE POINT IN THE GRID ONE SQUARE FARTHER
*           FROM THE RIVER MOUTH THAN CURRENT POINT
*   XINC -- SCALED DISTANCE BETWEEN POINTS REPRESETED BY HEIGHT
*           VECTOR (X INCREMENT)
*   XPAGE -- LENGTH OF THE FRAME(BOTTOM) FOR PLOT
*   XPOSTN -- DISTANCE FROM THE RIVER MOUTH TO CURRENT POSITION
*             (X POSITION). GIVEN IN METRES.

*   Y -- VECTOR OF POINTS REPRESENTING LATERAL POSITION IN THE RIVER.
*        USED IN CALCULATING HEMIPELAGIC FLUX.
*   YEARX -- DSTANCE FROM  X-AXIS IN  PLOT WHERE
*            THE YEAR MESSAGE IS TO BE PRINTED
*   YEARY -- DSTANCE FROM  Y-AXIS IN  PLOT WHERE
*            THE YEAR MESSAGE IS TO BE PRINTED
*   YEARS -- NUMBER OF YEARS THE MODEL IS TO BE RUN WITH CURRENT
*            SET OF PARAMETERS
*   YINC -- SCALED DISTANCE(METRES) BETWEEN LABELLED POINTS ON
*           ORDINATE AXIS OF PLOT
*   YPAGE -- LENGTH OF FRAME(SIDE) OF PLOT(INCHES)
*   YPOSTN -- LATERAL DISTANCE FROM CENTRE OF PLUME (METRES)
*   YRCOR -- FACTOR THAT CONVERTS TOTAL SEDIMENT DEPOSITED
*            IN A CONSTANT NUMBER OF ITERATIONS TO A SEDIMENTATION
*            RATE IN CM/YR
*   YR2SEC -- NUMBER OF SECONDS IN A YEAR
*
**************************************************************************
*
*  PSEUDO CODE FOR MAIN LINE
*
*  INITIALIZE CONSTANTS
*  GET DIFFUSION COEFFICIENT
*  WHILE THE DIFFUSION COEFFICIENT >= 0
*     [-
*     [ GET A SET OF PARAMETERS
*     [ FIND THE RIVER MOUTH
*     [ WHILE THERE ARE MORE PROFILES TO OUTPUT FOR THIS SET OF PARMS
*     [    [-
*     [    [ DO THE COMPUTATIONS FOR THIS TIME STEP (ITRATE)
*     [    [ ADJUST SEA LEVEL
*     [    [ IF THE RIVER MOUTH HAS FALLEN BELOW SEALVL
*     [    [    [-
*     [    [    [ REGRESS THE MODEL
*     [    [    [-
*     [    [ ELSE
*     [    [    [-
*     [    [    [  IF THE FIRST BIN HAS FILLED TO SEALVL
*     [    [    [    [-
*     [    [    [    [ PROGRADE THE MODEL
*     [    [    [    [-
*     [    [    [-   [-
*     [    [ IF IT IS TIME TO OUTPUT A PROFILE
*     [    [    [-
*     [    [    [ FILE THE CURRENT PROFILE
*     [    [    [ INCREMENT PROFILE COUNT
*     [    [-   [-
*     [ REINITIALIZE PROFILE COUNT
*     [ GET DIFFUSION COEFFICIENT
*     [-
*  END
```

```
*
*************************************************************
*
*  PSEUDO CODE FOR ITRATE
*
*  INCREMENT TIME
*  COMPUTE BYPASS
*  IF THE FORESLOPE IS TOO SHALLOW FOR BYPASSING
*      [-
*      [DUMP <- ENTIRE BEDLOAD
*      [-
*  ELSE
*      [-
*      [DUMP <- BEDLOAD NOT BYPASSED
*      [-
*  DEPOSIT DUMP,BYPASS,FLX IN AREA WHERE DUMPING OCCURS
*  DEPOSIT BYPASS AND FLX OVER REST OF PROFILE
*  IF DIFFUSION COEFICIENT > 0
*      [-
*      [DIFFUSE
*      [-
*  IF IT IS TIME TO OUTPUT DETAILS
*      [-
*      [OUTPUT DETAILS
*      [-
*  IF IT IS TIME TO OUTPUT RATES CHECKED AT TIME INTERVALS
*      [-
*      [OUTPUT RATES
*      [INCREMENT TIME FOR NEXT OUTPUT
*      [-
*  RETURN
*********************************************************************
*

      REAL LEVEL,SEAINT
      LOGICAL PRINT
*     VARIABLES FROM COMMON BLOCK GLOBAL
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
      INTEGER START,PTS,ENDDMP,LOWPT
      LOGICAL TIMCTL,QUIT
*    VARIABLES FROM COMMON BLOCK PRTCNT
      REAL TOFTEN,DOFTEN,NXTPRT
      INTEGER PROFLS,PRTCNT
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      COMMON /PRTCTL/ TOFTEN,DOFTEN,PROFLS,PRTCNT,NXTPRT
      LEVEL(T)=0.00

*________________________________________________________________________
* FILES
*
*      DETAIL -- STORES DETAILS ABOUT SEDIMENTATION RATE AT REQUESTED TIME
*
*      USER -- INPUT PARAMETERS FOR RUN
*
*      ADVANCE -- RECORD OF TIME OF EACH MODEL MOVE IE, PROGRADATION
*                 OR REGRESSION
*
*      TRACK -- RECORD OF SEDIMENTATION RATES AND SLOPE AT REQUESTED
*               POINTS, RECORDED AT INTERVALS THROUGHOUT THE RUN.
*
*      DELFILE -- THIS FILE IS INTENDED AS THE OUTPUT FILE FOR THE
*                 JOB. THE PROGRAM WRITES AN ECHO OF PARAMETERS,
*                 TRACKING INFORMATION, AND PROGRADATION RATES TO
*                 THIS FILE. THE JOB APPENDS DETAIL AND THE OUTPUT
*                 FROM GRAIN TO IT.
*
*      PLFILE -- GENERATED BY THE DISSPLAY ROUTINES (PLOT FILE)
*________________________________________________________________________
      OPEN(10,FILE='DETAIL',STATUS='NEW')
```

```
      OPEN(11,FILE='USER',STATUS='OLD')
      OPEN(13,FILE='ADVANCE',STATUS='NEW')
      OPEN(15,FILE='TRACK',STATUS='NEW')
      OPEN(20,FILE='DELFILE',STATUS='NEW')

*     READ/INITIALIZE CONSTANTS FOR RUN
      CALL INIT(SEAINT)

*     GET THE DIFFUSION COEFFICIENT.
      READ(11,*,END=9900,ERR=9901)
      READ(11,*,END=9900,ERR=9901)DIFSN
      WRITE(20,'(''DIFFUSION '',F12.4)')DIFSN
      QUIT = .FALSE.
*    WHILE THE DIFFUSION COEFFICIENT >= 0, COLLECT A SET OF PARMS AND
*    RUN THE MODEL
1000  CONTINUE
        IF (DIFSN .LT. 0.0) GOTO 2000
*       GET THE PARAMETERS FOR THIS RUN THROUGH THE LOOP.
*       INITIALIZE CONSTANTS AND VARIABLES FOR THE PARAMETER SET.
        CALL GETPMS
*       SET START(FIND THE RIVER MOUTH)
        CALL SETST
        LOWPT = START
*       INIT TIME OF NEXT PROFILE OUTPUT.
        CALL FSTPRT

*       WHILE THERE ARE MORE PROFILES TO PRINT, CONTINUE TO RUN WITH
*       THIS SET OF PARMS
1200    CONTINUE
          IF(PRTCNT .GE. PROFLS) GOTO 1300
          PRINT = .FALSE.
*         CALCULATIONS FOR NEXT TIME STEP
          CALL ITRATE
*         (QUIT = TRUE) => TRIED TO DIFFUSE BEDROCK. END RUN.
          IF (QUIT) GOTO 99999

*         ADJUST THE SEALVL AND CHECK TO SEE IF PROGRADATION OR
*         REGRESSION IS NECESSARY
          SEALVL = SEAINT + LEVEL(TIME*TIMCOR)/UNTHGT
*         IF THE FIRST POINT PAST THE RIVER MOUTH IS VERY NEAR
*         SEALVL, MOVE THE MODEL FORWARD
          IF((H(START) .LT. (1.0-2.0*TOL)*SEALVL) .OR.
     C             (H(LOWPT) .LT. (1.0-2.0*TOL)*SEALVL)) THEN
            CALL REGRAD
          ELSE IF(H(START+1) .GT. (1.0-TOL)*SEALVL) THEN
            CALL PROGRD(PRINT)
          ENDIF
*         (QUIT = TRUE) => TRIED TO MOVE BEYOND SYSTEM BOUNDARIES
          IF(QUIT)GOTO 99999

*         IF CONTROL IS BY TIME AND IT IS TIME TO PRINT OUT A PROFILE,
*         INCREMENT TIME OF NEXT PRINT AND SET ON FLAG
          IF ((TIMCTL) .AND. (TIME .GE. NXTPRT)) THEN
            NXTPRT = NXTPRT + TOFTEN
            PRINT = .TRUE.
          ENDIF

*         IF WE SHOULD PRINT NOW, INCREMENT COUNT AND CALL PLOTTING
*         ROUTINE
          IF (PRINT) THEN
            PRTCNT = PRTCNT + 1
            CALL FILEIT
          ENDIF
          GOTO 1200
1300    CONTINUE

*       GET THE NEXT DIFFUSION COEFFICIENT. (DIFSN < 0) => QUIT.
        READ(11,*,END=9900,ERR=9901)
        READ(11,*,END=9900,ERR=9901)DIFSN
        WRITE(20,'(''DIFFUSION '',F12.4)')DIFSN
        PRTCNT = 0
```

```
          GOTO 1000

2000   CONTINUE

       GOTO 99999
9900   WRITE(20,'(''NO RECORD IN USER'')')
       GOTO 99999
9901   WRITE(20,'(''ERROR READING USER'')')
99999 CONTINUE
       CALL ENDPL(0)
       CALL DONEPL
       CALL CLENUP
       CLOSE (9)
       CLOSE (10)
       CLOSE (11)
       CLOSE (13)
       CLOSE (15)
       CLOSE (20)
       STOP
       END
***********************************************************************

**************************************************************
* SUBROUTINE INIT -- INITIALIZES PLOTTING ROUTINES.
* COLLECTS THE PARAMETERS THAT  REMAIN FIXED FOR THE ENTIRE MODEL RUN.
* INITIALIZES THE CONSTANTS.
       SUBROUTINE INIT(SEAINT)
       REAL DG2RAD,DEGREE,XINC,DMPLGT,INTSED,SEAINT
       CHARACTER CNTRL
*      VARIABLES FOR COMMON BLOCK, ITRINT
       REAL SLOPEX,TRACKX,FLOOR(1001),DETIME
       INTEGER TRKPT,BINS
*      VARIABLES FOR COMMON BLOCK, BYINT
       REAL TANMAR
*      VARIABLES FOR COMMON BLOCK PRTCTL
       INTEGER PRTCNT,PROFLS
       REAL TOFTEN,DOFTEN,NXTPRT
*      VARIABLES FOR COMMON BLOCK,GLOBAL
       INTEGER START,PTS,ENDDMP,LOWPT
       REAL DIFSN,TOL
       REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
       LOGICAL TIMCTL,QUIT
       COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
      C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
       SAVE /GLOBAL/
       COMMON /ITRINT/ SLOPEX,TRACKX,TRKPT,FLOOR,DETIME,BINS
       COMMON /BYINT/ TANMAR
       COMMON /PRTCTL/ TOFTEN,DOFTEN,PROFLS,PRTCNT,NXTPRT
       SAVE /ITRINT/
       SAVE /BYINT/
       SAVE /PRTCTL/
       DATA INTSED/0.1/

       DG2RAD = 3.141592654/180.0
       BINS = 1000
       PTS = BINS + 1
       XINC = 1.0/BINS
       TIME = 0.0
       PRTCNT = 0
       TOL = 0.01


*      READ PAST COMMENT LINE
       READ(11,*,END=9900,ERR=9900)
       READ(11,*,END=9900,ERR=9900)UNTHGT
       WRITE(20,'(''UNIT HEIGHT '',F8.1,'' M'')')UNTHGT
*      ADJUST TO ACCOUNT FOR 0.1% ADDED WHEN WE PUT IN WHILE ESTABLISHING
*      THE INITIAL PROFILE
       UNTHGT = UNTHGT / 0.999
*      COMPUTE THE DIMENSIONLESS STARTING  HEIGHT
*      AND THE HEIGHT OF THE BEDROCK(FLOOR)
```

```
      CALL BOTTOM
      DO 100 I=1,PTS
         FLOOR(I) = H(I) - INTSED
100   CONTINUE


      READ(11,*,END=9900,ERR=9901)LENGTH
      WRITE(20,'(''DST TO SILL'',F8.1,'' KM'')')LENGTH
*     CONVERT TO METRES
      LENGTH = LENGTH *1000.0
      DSTINC = LENGTH / BINS
      TIMCOR = LENGTH * LENGTH
      READ(11,*,END=9900,ERR=9901)WIDTH
      WRITE(20,'(''WIDTH OF FJORD BOTTOM'',F8.1,'' KM'')')WIDTH
*     CONVERT TO METRES
      WIDTH = WIDTH * 1000.0
      READ(11,*,END=9900,ERR=9901)SLOPEX
      WRITE(20,'(''TRACK SLOPE AT X ='',F8.1,'' KM'')')SLOPEX
      READ(11,*,END=9900,ERR=9901)DETIME
      WRITE(20,'(''SEDIMENT DETAILS AT ABOUT'',F7.1,'' YRS'')')DETIME
*     SCALE THE TIME
      DETIME = DETIME/TIMCOR
      READ(11,*,END=9900,ERR=9901)TRACKX
      WRITE(20,'(''TRACK ACCUMULATION AT X ='',F8.1,''KM'')')TRACKX
*     CONVERT TO METRES AND FIND THE INDEX THAT REPRESENTS
*     THE TRACKED POINT.
      TRKPT = 1 + INT(TRACKX*1000.0/DSTINC)
      WRITE(15,'(2F6.1)')TRACKX,SLOPEX
      SLOPEX = SLOPEX*1000.0

      READ(11,*,END=9900,ERR=9901)DMPLGT
*     ROUND TO A MULTIPLE OF DSTINC AND CONVERT TO NUMBER OF POINTS
*     OVER WHICH TO DUMP.
      ENDDMP = INT(0.5 +DMPLGT/DSTINC)
      DMPLGT = DSTINC * ENDDMP
*     HAVE TO DUMP AT AT LEAST TWO POINTS
      IF (ENDDMP .LT. 2) THEN
        DMPLGT = 2.0*DSTINC
        ENDDMP = 2
      ENDIF

      WRITE(20,'(''DST OVER WHICH TO DUMP BEDLOAD'',F7.1,'' M'')')DMPLGT
      READ(11,'(A)',END=9900,ERR=9901)CNTRL
      IF (CNTRL .EQ. 'T') THEN
        TIMCTL = .TRUE.
        WRITE(20,'(''CONTROL BY TIME'')')
      ELSE IF (CNTRL .EQ. 'D') THEN
        TIMCTL = .FALSE.
        WRITE(20,'(''CONTROL BY DISTANCE PROGRADED'')')
      ENDIF
      READ(11,*,END=9900,ERR=9901)DEGREE
      WRITE(20,'(''SLOPE WHEN BYPASS STARTS:'',F5.1,''DEG'')')DEGREE
*     COMPUTE THE POSITIVE TAN OF THE SLOPE AND SCALE
      TANMAR = DSTINC * TAN(DG2RAD*DEGREE) / UNTHGT
      READ(11,*,END=9900,ERR=9901)SEAINT
      WRITE(20,'(''INITIAL DEPTH OF BASIN: '',F8.2,'' M'')')SEAINT

*     CONVERT TO DIMENSIONLESS HEIGHT. ADJUST TO ACCOUNT FOR 0.1% ADDED
*     WHEN PUTTING IN THE EXPONENTIAL BOTTOM.
      SEAINT = SEAINT/UNTHGT  +   0.001
      SEALVL = SEAINT


*      INITIALIZE THE PLOTTING ROUTINE AND OUTPUT INITIAL PROFILE
       CALL INTPLT
       CALL FILFST


       GOTO 99999
9900   WRITE(20,'(''NO RECORD IN USER'')')
9901   WRITE(20,'(''ERROR READING USER'')')
```

```
99999  CONTINUE
      RETURN
      END

************************************************************************


* SUBROUTINE SEDFLX

*    USES A VELOCITY FIELD MODEL TO COMPUTE SUSPENDED SEDIMENT DEPOSITION
*    UNDER A RIVER PLUME. CALCULATES AVERAGE DEPTH OF SEDIMENT IN
*    CROSS-SECTIONS AT INTERVALS DOWNSTREAM FROM FJORD HEAD.
*    SCALES THE FLX RATES.
*
      SUBROUTINE SEDFLX(D)

      REAL XPOSTN,Y(201),VX(201),VY(201),T(201)
      REAL YR2SEC,D
      INTEGER XBPNT,I
*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
      LOGICAL TIMCTL,QUIT
*     VARIABLES FOR COMMON BLOCK, FLXDWN
      REAL FLXTRM(2001)
*     VARIABLES FOR COMMON BLOCK,  FINDVX
      REAL CHANNL,V0,XB
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/
      COMMON /FLXDWN/ FLXTRM
      SAVE /FLXDWN/
      COMMON /SED/ T,Y,VX,VY
      COMMON /FINDVX/ CHANNL,V0,XB
      SAVE /FINDVX/
*
*     COMPUTE THE INDEX OF THE POINT THAT REPRESENTS XB
      XBPNT = INT(XB/DSTINC) + 1

      YR2SEC = 3600.0 * 24.0 * 365.0

*    INITIALIZE THE FLX VECTOR
      DO 100 I= 1,PTS
         FLXTRM(I) = 0.0
100   CONTINUE
      Y(1) = 0.0
      VX(1) = V0
      VY(1) =  0.0
      T(1) = 0.0
      DO 200 I=2,201
        Y(I) = Y(I-1) + CHANNL/400.0
        VX(I) = V0
        VY(I) = 0.0
        T(I) = 0.0
200   CONTINUE

*     COMPUTE FLX IN THE NEAR ZONE (ZONE UNDER SPREADING PLUME)
      XPOSTN = DSTINC
      DO 500 I=2,XBPNT
        FLXTRM(I) = YR2SEC*FLXIN(XPOSTN)/WIDTH
        XPOSTN = XPOSTN + DSTINC
500   CONTINUE

*     COMPUTE FLX IN THE FAR ZONE (AFTER PLUME ATTAINS MAX WIDTH)
      DO 600 I= XBPNT+1,PTS
        FLXTRM(I) = YR2SEC * FLXOUT(XPOSTN)/WIDTH
        XPOSTN = XPOSTN + DSTINC
600   CONTINUE

*     SCALE THE FLUX RATE
```

```
      DO 700 I = 1,PTS
        FLXTRM(I) = D * FLXTRM(I)
700   CONTINUE
      RETURN
      END
*******************************************************************************

*    SUBROUTINE FLXIN
*    COMPUTES THE AREA (M**2/SEC) OF SUSPENDED SEDIMENT DEPOSITED UNDER A
*    SPREADING PLUME IN A CROSS-SECTION XPOSTN METRES FROM THE
*    RIVER MOUTH
*    THE PNTFLX(1),T(1),Y(1),VX(1),VY(1) ARE ASSOCIATED WITH THE COLLECTION
*    POINT AT THE CENTRE OF THE PLUME. THE VECTORS REPRESENT COLLECTION
*    POINTS IN HALF A LATERAL CROSS-SECTION OF A SYMMETRICAL PLUME.

      REAL FUNCTION FLXIN(XPOSTN)

      REAL XPOSTN,M,N,PNTFLX(201),XBACK,TINC,VYINC,XFRD,HALFUP
      REAL VXBACK,VXFRD
      INTEGER PTSACC,BINACC,GRAINS

*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
      LOGICAL TIMCTL,QUIT
*     VARIABLES FOR COMMON BLOCK GRAIN
      REAL CNTN(4), LMDA(4), DENSTY(4), DEPTH
*     VARIABLES FOR COMMON BLOCK SED
      REAL T(201),Y(201),VX(201),VY(201)
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/
      COMMON /GRAIN/ CNTN,LMDA,DENSTY,DEPTH
      SAVE /GRAIN/
      COMMON /SED/ T,Y,VX,VY
      SAVE /SED/

      DATA E,PTSACC,GRAINS/2.718281828,201,4/
      BINACC = PTSACC-1
      XBACK = XPOSTN - DSTINC
      XFRD = XPOSTN + DSTINC

*     FOR EACH COLLECTON POINT IN THE HALF CROSS-SECTION, CALCULATE THE
*     SEDIMENT DEPOSITED
      DO 100 I=1,PTSACC
*       TINC = TIME TO MOVE DSTINC METRES DOWN FJORD
        TINC = DSTINC/VX(I)
*       COMPUTE NEW LATERAL POSITION AND CUMULATIVE TIME
        Y(I) = Y(I) + TINC*VY(I)
        T(I) = T(I) + TINC
        PNTFLX(I) = 0.0
*       FOR EACH GRAIN SIZE, COMPUTE AND ADD THE FLX DEPOSITED AT
*       THIS COLLECTION POINT
        DO 50 J = 1,GRAINS
          M = LMDA(J)*CNTN(J)*DEPTH/DENSTY(J)
          N = -LMDA(J) * T(I)
*         AVOID UNDERFLOW ERROR THAT OCCURS IF POWER OF E VERY SMALL
          IF (N .GT. -675.0) THEN
            PNTFLX(I) = PNTFLX(I) + M * (E**N)
          ENDIF
50      CONTINUE

*       THERE IS NO LATERAL VELOCITY COMPONENT AT THE CENTER OF THE PLUME
*       IF I>1, CALCULATE THE LATERAL VELOCITY COMPONENT USING THE
*       APPROXIMATION:
*          (CHANGE IN VY)/(TSTEP) = - (CHANGE IN VX)/(TSTEP)
*          VX HERE IS COMPUTED BY FUNCTION FVNX
        IF (I .GT. 1) THEN
          HALFUP = (Y(I)+Y(I-1))/2.0
          VXBACK = FNVX(XBACK,HALFUP)
```

```
          VXFRD = FNVX(XFRD,HALFUP)
          VYINC = -(VXFRD - VXBACK)*(Y(I)-Y(I-1))/(2.0*DSTINC)
          VY(I) = VY(I-1) + VYINC
        ENDIF
        VX(I) = FNVX(XPOSTN,Y(I))
100     CONTINUE

*       INTEGRATE ACROSS THE HALF CROSS-SECTION AND DOUBLE TO OBTAIN THE
*       AREA OF SEDIMENT DEPOSITED. NOTE THAT NOT DIVIDING BY 2 WHEN
*       SUMMING FLXIN IS EQUIVALENT TO DOUBLING.
        FLXIN = 0.0
        DO 200 I=1,BINACC
          FLXIN = FLXIN + (PNTFLX(I) + PNTFLX(I+1)) * (Y(I+1)-Y(I))
200     CONTINUE
        RETURN
        END
*****************************************************************************

*    SUBROUTINE FLXOUT
*    COMPUTES THE AREA (M**2/SEC) OF SUSPENDED SEDIMENT DEPOSITED UNDER A
*    PLUME OF CONSTANT WIDTH IN A CROSS-SECTION XPOSTN METRES FROM THE
*    RIVER MOUTH
*    THE PNTFLX(1),T(1),Y(1),VX(1),VY(1) ARE ASSOCIATED WITH THE COLLECTION
*    POINT AT THE CENTRE OF THE PLUME. THE VECTORS REPRESENT COLLECTION
*    POINTS IN HALF A LATERAL CROSS-SECTION OF A SYMMETRICAL PLUME.
*    ASSUME VELOCITY REMAINS CONSTANT DOWN FJORD. THAT IS VELOCITY
*    IS A FUNCTION OF DISTANCE FROM CENTRE.

      REAL FUNCTION FLXOUT(XPOSTN)

      REAL M,N,PNTFLX(201),E,XPOSTN
      INTEGER BINACC,PTSACC,I,J,GRAINS

*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
      LOGICAL TIMCTL,QUIT
*     VARIABLES FOR COMMON BLOCK GRAIN
      REAL CNTN(4), LMDA(4), DENSTY(4), DEPTH
*     VARIABLES FOR COMMON BLOCK SED
      REAL T(201),Y(201),VX(201),VY(201)
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/
      COMMON /GRAIN/ CNTN,LMDA,DENSTY,DEPTH
      SAVE /GRAIN/
      COMMON /SED/ T,Y,VX,VY
      SAVE /SED/

       DATA E,PTSACC,GRAINS/2.718281828,201,4/
       BINACC = PTSACC - 1

*     FOR EACH COLLECTON POINT IN THE HALF CROSS-SECTION, CALCULATE THE
*     SEDIMENT DEPOSITED
       DO 100 I=1,PTSACC
*         INCREMENT THE CUMULATIVE TIME
          T(I) = T(I) + DSTINC/VX(I)
          PNTFLX(I) = 0.0
*         FOR EACH GRAIN SIZE, COMPUTE AND ADD THE FLX DEPOSITED AT
*         THIS COLLECTION POINT
          DO 50 J = 1,GRAINS
            M = LMDA(J)*CNTN(J)*DEPTH/DENSTY(J)
            N = -LMDA(J) * T(I)
*           AVOID UNDERFLOW ERROR THAT OCCURS IF POWER OF E VERY SMALL
            IF (N .GT. -675.0) THEN
              PNTFLX(I) = PNTFLX(I) + M * (E**N)
            ENDIF
*         IF (I.EQ.201)THEN
*         ENDIF
50        CONTINUE
```

```
100   CONTINUE

*      INTEGRATE ACROSS THE HALF CROSS-SECTION AND DOUBLE TO OBTAIN THE
*      AREA OF SEDIMENT DEPOSITED. NOTE THAT NOT DIVIDING BY 2 WHEN
*      SUMMING FLXOUT IS EQUIVALENT TO DOUBLING.
      FLXOUT = 0.0
      DO 200 I=1,BINACC
        FLXOUT = FLXOUT + (PNTFLX(I) + PNTFLX(I+1))*(Y(I+1)-Y(I))
200   CONTINUE
      RETURN
      END
*******************************************************************

*      FUNCTION FNVX

*      COMPUTES THE X-COMPONENT OF VELOCITY AT A POINT(XPOSTN,
*      YPOSTN) N A TWO DIMENSIONAL SPREADING RIVER PLUME.
*      THE ORIGIN IS THE CENTRE OF THE RIVER CHANNEL AT THE
*      DELTA HEAD. X IS LONGITUDINAL POSITION. Y IS LATERAL POSITION.
*      USES THE MODEL DEVELOPED BY ALBERTSON ET AL. (1950)
*

      REAL FUNCTION FNVX(XPOSTN,YPOSTN)
      REAL X0,XPOSTN,BORDER,YPOSTN,K,C,PI,E
*     VARIABLES FOR COMMON BLOCK,  FINDVX 3
      REAL CHANNL,V0,XB
      COMMON /FINDVX/ CHANNL,V0,XB
      SAVE /FINDVX/

      DATA C,PI,E/.109,3.14159265,2.718281828/

*      COMPUTE THE DISTANCE FROM THE RIVER MOUTH WHERE THE ZONE OF
*      ESTABLISHMENT ENDS.
       X0 = CHANNL / (SQRT(PI)*C)

*      IF WE ARE WITHIN THE ZONE OF ESTABLISHMENT,FIND THE BORDER ON THE
*      PLUG FLOW AREA
       IF(XPOSTN.LT.X0)THEN
         BORDER = (1.0-XPOSTN/X0)*CHANNL/2.0
*        IF OUTSIDE PLUG FLOW AREA USE APPROPRIATE EQUATIONS
         IF(YPOSTN.GT.BORDER)THEN
           K = -(YPOSTN + (XPOSTN*C*SQRT(PI) - CHANNL)/2.0 )**2.0
           K = K/(2*C*C*XPOSTN*XPOSTN)
           FNVX = V0* (E**K)
*        IF INSIDE PLUG FLOW AREA, VELOCITY IS CONSTANT
         ELSE
           FNVX = V0
         ENDIF
*      IF IN THE ZONE OF ESTABLISHED FLOW, USE APPROPRIATE EQUATIONS.
       ELSE
         K = -(YPOSTN*YPOSTN/(2.0*C*C*XPOSTN*XPOSTN))
         FNVX = V0 * SQRT(X0/XPOSTN) * (E**K)
       ENDIF
       RETURN
       END

*************************************************************************

*      SUBROUTINE ITRATE
*      CALCULATES AND ADDS THE SEDIMENT DEPOSITED IN ONE TIMESTEP.
*      DIFFUSES THE SEDIMENT.
*      CHECKS IF IT IS TIME TO OUTPUT A PROFILE, SEDIMENTATION DETAILS
*      OR DATA ABOUT ONE OF THE POINTS BEING TRACKED. TAKES APPROPRIATE
*      ACTION.
*      QUIT IS SET TO TRUE IF THERE IS AN ATTEMPT TO DIFFUSE THE BEDROCK

      SUBROUTINE ITRATE

      REAL OLDH, TANSLP, DG2RAD, OUTSED, OUTSRC, OUTTOT, OLD(1001)
      REAL HLAST,HTEMP,TRKOFT,PTSLP
      INTEGER DETCNT, TRKCNT, ENDCNT, SLPPT, NUMPTS, OFTEN
```

```
      LOGICAL DETPRP,TRKPRP
*     VARIABLES FOR COMMON BLOCK, FLXDWN
      REAL FLXTRM(2001)
*     VARIABLES FOR COMMON BLOCK, BYDWN
      REAL BYPASS(1001)
      LOGICAL SHALOW
*     VARIABLES FOR COMMON BLOCK, ITRINT
      REAL SLOPEX,TRACKX,FLOOR(1001),DETIME
      INTEGER TRKPT,BINS
*     VARIABLES FOR COMMON BLOCK, ITRCST
      REAL TSTEP,YRCOR,DMPALL,QDUMP,A,B,D
*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
      LOGICAL TIMCTL,QUIT
      COMMON /ITRINT/ SLOPEX,TRACKX,TRKPT,FLOOR,DETIME,BINS
      SAVE /ITRINT/
      COMMON /FLXDWN/ FLXTRM
      SAVE /FLXDWN/
      COMMON /BYDWN/ BYPASS,SHALOW
      SAVE /BYDWN/
      COMMON /ITRCST/ TSTEP, YRCOR,DMPALL,QDUMP,A,B,D
      SAVE /ITRCST/
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/
      SAVE DETPRP,TRKPRP,OLDH,OLD,DETCNT,TRKCNT,TRKTIM

      DATA DETPRP,TRKPRP/.FALSE.,.FALSE./
      DATA ENDCNT,TRKOFT,TRKTIM/10,200.0,0.0/
      DATA NUMPTS,TRKCNT,DETCNT/30,-10,-10/

*_____________________________________________________________________________
*      DEPOSIT FLX, BYPASS AND DUMP MATERIAL
         SHALOW = .FALSE.
         TIME = TIME + TSTEP
         CALL BYPAS
         IF (SHALOW) THEN
*             TOO SHALLOW TO BYPASS - DUMP ALL OF QB
              QDOWN = DMPALL
         ELSE
*              DUMP THE PORTION OF QB NOT BYPASSED
               QDOWN = QDUMP
         ENDIF

            DO 1010 I = START+1, START+ENDDMP
               H(I) = H(I) + QDUMP + FLXTRM(I) + BYPASS(I)
1010        CONTINUE

            DO 1020 I = START+ENDDMP+1, PTS
               H(I) = H(I) + FLXTRM(I) + BYPASS(I)
1020        CONTINUE

*_____________________________________________________________________________
*       IF THE DIFFUSION COEFFICIENT IS NOT 0, DIFFUSE THE SEDIMENT.
*       START AND PTS ARE BOUNDARY POINTS.
*       IF DIFFUSION INVOLVES CUTTING INTO BEDROCK, STOP THE MODEL
            IF(DIFSN .GT. 1.0) THEN
*             DIFFUSE
              HLAST = H(START)
              H(START) = 2.0*A*H(START+1) + B*H(START)
              DO 1100 I = START+1, BINS
                 HTEMP = H(I)

                 H(I) = A*(HLAST + H(I+1)) + B*H(I)
                  IF(H(I).LT.FLOOR(I)) THEN
                     WRITE(20,'(''TRYING TO DIFFUSE FLOOR'')')
                     QUIT = .TRUE.
                     GOTO 99999
                  ENDIF
```

```
                  HLAST = HTEMP
1100       CONTINUE

           HTEMP = H(PTS)
           H(PTS) = 2.0*A*HLAST + B*H(PTS)
       ENDIF

*____________________________________________________________________________
*       IS THIS THE TIME FOR WHICH DETAILS HAVE BEEN REQUESTED?
*       IF SO, COMPUTE THE CHANGE OVER A NUMBER(ENDCNT) OF ITERATIONS
*       THEN OUTPUT THE DETAILS. THE CHANGE IN HEIGHT IS AVERAGED
*       TO MODERATE THE VARIATIONS THAT OCCUR FROM ONE TIME STEP TO THE
*       NEXT.

*        IF WE'VE STORED OLD DETAILS AND ARE COUNTING THROUGH
*        ITERATIONS, INCREASE THE COUNT
         IF (DETPRP) THEN
             DETCNT = DETCNT + 1
         ENDIF

*        IF IT IS TIME TO PRINT OUT DETAILS, MAKE SURE WE NEVER HIT
*        DETTIM AGAIN,, INIT COUNT OF ITERATIONS, SET ON DETAIL
*        PREP FLAG, STORE CURRENT HEIGHT.
         IF(TIME .GE. DETIME) THEN
            DETIME = 10.0**8
            DETCNT = 0
            DETPRP = .TRUE.
            DO 1200 I = 1,PTS
               OLD(I) = H(I)
1200        CONTINUE
         ENDIF

*        IF WE'VE COUNTED THROUGH THE REQUIRED NUMBER OF ITERATIONS
*        SINCE SAVING THE OLD HEIGHT, RECORD ACCUMULATION RATES.
         IF(DETCNT .EQ. ENDCNT) THEN
            DETCNT = -10
            DETPRP = .FALSE.
            WRITE(10,'(''ACCUMULATION RATES AFTER'',F8.1,
     C      '' YEARS'')')TIME*TIMCOR
            WRITE(10,'('' '')')
            WRITE(10,'(''DISTANCE HEMIPELAGIC '',
     C      ''HEMIPELAGIC TOTAL'')')
            WRITE(10,'(''(KM)     RAIN'',8X,''RAIN AND'',
     C      4X,''ACCUMULATION'')')
            WRITE(10,'(9X,''(CM/A)'',6X,''BYPASS'',6X,''(CM/A)'')')
            WRITE(10,'(21X,''(C/A)'')')

            OFTEN = INT((PTS-START)/NUMPTS)
            DO 1250 I = START+1,PTS,OFTEN
*              PRINT RATES. *100 BECAUSE CHANGING M TO CM.
               WRITE(10,'(F5.1,F9.1,2F12.1)')(I-START)*DSTINC/1000.0,
     C         YRCOR*100.0*FLXTRM(I),
     C         YRCOR*100.0*(BYPASS(I)+FLXTRM(I)),
     C         YRCOR*100.0*(H(I)-OLD(I))/ENDCNT
1250       CONTINUE
          ENDIF

*____________________________________________________________________________
*       IS THIS A TIME FOR WHICH WE WANT TO PRINT OUT SEDIMENTATION
*       RATES AND SLOPE FOR THE POINTS BEING TRACKED?
*       IF SO, COMPUTE THE CHANGE OVER A NUMBER(ENDCNT) OF ITERATIONS
*       THEN OUTPUT THE DETAILS. THE CHANGE IN HEIGHT IS AVERAGED
*       TO MODERATE THE VARIATIONS THAT OCCUR FROM ONE TIME STEP TO THE
*       NEXT.
*       SEDIMENTATION RATES ARE TRACKED AT A FIXED POINT THAT IS AN
*       INPUTTED DISTANCE FROM THE ORIGINAL RIVER MOUTH.
*       SLOPE IS TRACKED AT A FIXED DISTANCE FROM RIVER MOUTH. THE
*       POINT AT WHICH SLOPE IS TRACKED MOVES AS THE DELTA PROGRADES.

*        IF WE'VE STORED THE OLD HEIGHT AND ARE JUST COUNTING UP
*        TO THE REQUIRED NUMBER OF ITERATIONS, INCREASE THE COUNT.
```

```
          IF (TRKPRP)THEN
            TRKCNT = TRKCNT+1
          ENDIF

*         IF IT IS TIME TO STORE ACCUMULATION AND SLOPE DETAILS,
*         CALCULATE TIME FOR NEXT TRACKING OUTPUT, RECORD HEIGHT, INIT COUNT
*         OF ITERATIONS, SET ON FLAG THAT WE'RE PREPARING TO STORE
*         TRACKING INFO.
*         SLOPE DETAILS,
          IF(TIME.GE.TRKTIM)THEN
            TRKCNT = 0
            TRKTIM = TRKTIM + TRKOFT/TIMCOR
            OLDH = H(TRKPT)
            TRKPRP = .TRUE.
          ENDIF

*         IF WE'VE GONE THROUGH THE REQUIRED NUMBER OF ITERATIONS SINCE
*         STORING HEIGHT, RECORD ACCUMULATION AND SLOPE
*         NOTE *100 = CONVERT FROM M TO CM
          IF(TRKCNT.EQ.ENDCNT)THEN
            DG2RAD = 3.141592654/180.0
            TRKPRP = .FALSE.
            TRKCNT = -10

*        COMPUTE SLOPE SLOPEX METRES FROM RIVER MOUTH.
            SLPPT = START +INT(SLOPEX/DSTINC)
*           IF SLPPT IS NOT AT OR BEYOND THE SILL, FIND THE SLOPE
*           OTHERWISE SET SLOPE AT -95 AS A FLAG
            IF(SLPPT .LT. PTS)THEN
              TANSLP = (H(SLPPT-1)-H(SLPPT+1))*UNTHGT/(2.0*DSTINC)
              OUTSLP = ATAN(TANSLP) / DG2RAD
            ELSE
              OUTSLP = -95.0
            ENDIF

*           IF THE POINT WE'RE TRACKING IS STILL IN THE FOREDELTA OR
*           PRODELTA, COMPUTE SEDIMENTATION RATES AT POINT TRKPT
*           OTHERWISE,SET VALUES TO -95 AS FLAGS
            IF(TRKPT .GT. START)THEN
              OUTSED = YRCOR * FLXTRM(TRKPT)  * 100.0
              OUTSRC = YRCOR*BYPASS(TRKPT)*100.0 + OUTSED
              OUTTOT = YRCOR * (H(TRKPT)-OLDH) * 100.0 / (ENDCNT*1.0)
              TANSLP = (H(TRKPT-1)-H(TRKPT+1))*UNTHGT/(2.0*DSTINC)
              PTSLP = ATAN(TANSLP) / DG2RAD
            ELSE
              OUTSED = -95
              OUTSRC = -95
              OUTTOT = -95
              PTSLP = -95
            ENDIF
            WRITE(15,'(F12.1,5F12.8)')TIME*TIMCOR,OUTSLP,PTSLP,OUTSED,
     C           OUTSRC,OUTTOT
          ENDIF


99999     CONTINUE
          RETURN
          END


*************************************************************************

*      SUBROUTINE BYPAS
*      IF SLOPE IS STEEP ENOUGH FOR BYPASSING,
*         - ELEMENTS OF BYPAS <- 0 UNTIL SLOPE = DEGREE, AT X',SAY
*         - FROM X=X' TO X=X' + RUNOUT,
*           ELEMENTS OF BYPAS <- A VALUE SO THAT AREA DEPOSITED
*           IS TOTAL BYPASS MATERIAL / WIDTH OF FJORD,
*           BYPASS DEPOSITED AT X' IS TWICE THE THICKNESS OF
*           BYPASS DEPOSITED AT X'+RUNOUT.
*           IF THE SLOPE IS TOO SHALLOW FOR BYPASSING, SETS ALL
```

```
*          ELEMENTS TO 0 AND TURNS ON A FLAG.

      SUBROUTINE BYPAS

      REAL SPREAD,HGTCHG,HTCHG2,HTCHG3,P,POSTN
      INTEGER ENDRUN
*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
      LOGICAL TIMCTL,QUIT
*     VARIABLES FOR COMMON BLOCK, BYIN
      REAL RUNOUT,QBYPAS
      INTEGER  RUNNUM
*     VARIABLES FOR COMMON BLOCK, BYINT
      REAL TANMAR
*     VARIABLES FOR COMMON BLOCK, BYDWN
      REAL BYPASS(1001)
      LOGICAL SHALOW
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/
      COMMON /BYIN/ RUNOUT,RUNNUM,QBYPAS
      SAVE /BYIN/
      COMMON /BYINT/ TANMAR
      SAVE /BYINT/
      COMMON /BYDWN/ BYPASS,SHALOW

*     CHECK IF THE SLOPE OF THE FORESET IS STEEP ENOUGH TO CAUSE
*     BYPASSING
      HGTCHG = (H(START+ENDDMP) - H(START+ENDDMP+2) )/2.0
      HTCHG2 = ( H(START+ENDDMP-1) - H(START+ENDDMP+1) ) /2.0
      HTCHG3 = ( H(START+ENDDMP+1) - H(START+ENDDMP+3) ) /2.0
      IF(((HGTCHG.GE.TANMAR).OR.
     C      ((HTCHG2.GE.TANMAR).OR.(HTCHG3.GE.TANMAR)))
     C     .AND. (START .LT. PTS-1)) THEN
         SHALOW = .FALSE.
         BYPASS(START) = 0.0
         I = START + 1
         HGTCHG = (H(START) - H(I+1)) / 2.0

*        WHILE THE SLOPE IS GREATER THAN THE SLOPE MARGIN, OR WE ARE NEAR
*        THE TOP OF THE DELTA, NO BYPASSING MATERIAL IS DEPOSITED.
*
3100     CONTINUE
         IF((HGTCHG .LT. TANMAR) .AND.
     C      ((H(I)-H(PTS))  .LT. (0.7 * (H(START)-H(PTS))))) GOTO 3150
            BYPASS(I) = 0.0
            I = I + 1
            HGTCHG = (H(I-1) - H(I+1)) / 2.0
            GOTO 3100
3150      CONTINUE


         ENDRUN = RUNNUM + I

*        THE MATERIAL CANNOT BE SPREAD BEYOND THE SILL. IF WE'RE TOO
*        CLOSE TO THE SILL, SPREAD ONLY TO THE SILL
         IF (ENDRUN .LE. PTS) THEN
            SPREAD = RUNOUT
         ELSE
            ENDRUN = PTS
            SPREAD =  (PTS-I) * DSTINC
         ENDIF

         P = 2.0*QBYPAS/(3.0*SPREAD)
*        DEPOSIT THE BYPASS MATERIAL SO THAT THERE IS TWICE AS MUCH
*        AT THE RIVER END OF THE DEPOSIT AS AT THE SILL END OF THE
*        DEPOSIT.

         POSTN = 0.0
```

```
         DO 3200 I = I,ENDRUN
            BYPASS(I) = P * (2.0 - POSTN/SPREAD)
            POSTN = POSTN + DSTINC
3200     CONTINUE

         IF(ENDRUN .LT. PTS) THEN
            DO 3300 I = I,PTS
               BYPASS(I) = 0.0
3300        CONTINUE
         ENDIF

      ELSE

*       (IF TOO SHALLOW)
*        PRINT A WARNING AND SET ON FLAG
         WRITE(20,'(''TOO SHALOW AT RIVER MOUTH TO BYPASS'')')
         SHALOW = .TRUE.
         DO 3400 I = START,PTS
            BYPASS(I)=0.0
3400     CONTINUE

      ENDIF
      RETURN
      END
***************************************************************************

*     SUBROUTINE BOTTOM
*     ESTABLISHES THE HEIGHT OF THE INITIAL PROFILE ABOVE BASE.
*     SET UP SO THE THE DEEPEST POINT(AT THE SILL) IS 0.1% OF
*     UNTHGT.
*     IF YOU'RE CHANGING THE SHAPE OF THE BOTTOM, CHANGE THE
*     ADJUSTMENT OF UNTHGT AND SEALVL IN SUBROUTINE INIT AS THESE
*     CORRECT THE .1% ERROR. ALSO, THE PLOTTING ROUTINES,
*     FILFST AND FILEIT MAKE THE CORRECTION AND WILL HAVE TO BE CHANGED.
*
      SUBROUTINE BOTTOM

      REAL X,E,K,XINC
*     VARIABLES FOR COMMON BLOCK,GLOBAL 6
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL DIFSN,TOL
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      LOGICAL TIMCTL,QUIT
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/
      DATA E,K/2.718281828,-6.90775527898/

      XINC = 1.0 / (PTS - 1)
      X = 0.0
      DO 100 I=1,PTS
       H(I) = E**(K*X)
       X = X + XINC
100   CONTINUE

      RETURN
      END

*************************************************************************

*     SUPROUTINE INTPLT -- INITIALIZES THE PLOTTING ROUTINES AND THE
*     X-COORDINATES OF THE POINTS TO BE GRAPHED

      SUBROUTINE INTPLT

      REAL XPAGE, YPAGE,  TOPY, TOPX, HITE
      INTEGER I
*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
```

```
      LOGICAL TIMCTL,QUIT
*     VARIABLES FOR COMMON BLOCK, PLT
      REAL X(1001), LXAXIS,LYAXIS

      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/
      COMMON /PLT/ X,LXAXIS,LYAXIS
      SAVE /PLT/

      DATA XPAGE,YPAGE,HITE/18.0,7.0,0.07/

      TOPY = 1.1*UNTHGT
      TOPX = LENGTH / 1000.0
      XINC = 1.0 * INT(TOPX / 20.0)
      YINC = 1.0 * INT(TOPY / 5.0)
      LXAXIS = 14.0
      LYAXIS = 4.6
      CALL BGNPL(1)
      CALL COMPRS
      CALL PAGE(XPAGE,YPAGE)
      CALL AREA2D(LXAXIS,LYAXIS)
      CALL XNAME('DISTANCE FROM ORIGINAL RIVER MOUTH(KM)$',100)
      CALL YNAME('HEIGHT(M)$',100)
      CALL GRAF(0,XINC,TOPX,0,YINC,TOPY)
      CALL HEIGHT(HITE)

*     INITIALIZE THE VECTOR OF X-COORDINATES OF POINTS
      X(1) = 0.0
      DO 100,I=2,PTS
*        WANT OUTPUT IN KM
         X(I) = X(I-1) + DSTINC / 1000.0
100   CONTINUE

      RETURN
      END

****************************************************************************

*      SUBROUTINE FILFST -- PLOTS THE PROFILE OF THE INITIAL FLOOR
*      SCALED HEIGHTS ARE DIMENSIONED BEFORE OUTPUT.

       SUBROUTINE FILFST

       REAL YEARX,YEARY,MSGANG,Y(1001)
       INTEGER I
*
*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
      LOGICAL TIMCTL,QUIT
*     VARIABLES FOR COMMON BLOCK, PLT
      REAL X(1001),LXAXIS,LYAXIS

      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/
      COMMON /PLT/ X,LXAXIS,LYAXIS
      SAVE /PLT/

      DATA MSGANG/20.0/

           DO 100,I=1,PTS
*             .001 IS AN ADJUSTMENT BECAUSE OF THE EXTRA HEIGHT
*             ADDED IN MAKING AN EXPONENTIAL BOTTOM.
              Y(I)=UNTHGT*(H(I)- .001)
100        CONTINUE
           START = 1

           CALL CURVE(X,Y,1001,0)
```

```
*           COMPUTE X,Y DISTANCE FROM ORIGIN OF YEAR-OF-PROFILE MESSAGE
            YEARX = DSTINC * (START-1) * LXAXIS / LENGTH
            YEARY = LYAXIS
            CALL ANGLE(MSGANG)
            CALL REALNO(TIME*TIMCOR,1,YEARX,YEARY)
            CALL MESSAG('YEARS$',100,'ABUT','ABUT')

99999       CONTINUE

            RETURN
            END

***************************************************************************

*       SUBROUTINE FILEIT -- PLOTS CURRENT PROFILE. SCALED HEIGHTS ARE
*       DIMENSIONED BEFORE OUTPUT.

        SUBROUTINE FILEIT

        REAL YEARX,YEARY,MSGANG,Y(1001)
        INTEGER I
*
*      VARIABLES FOR COMMON BLOCK,GLOBAL
       INTEGER START,PTS,ENDDMP,LOWPT
       REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
       REAL DIFSN,TOL
       LOGICAL TIMCTL,QUIT
*      VARIABLES FOR COMMON BLOCK, PLT
       REAL X(1001),LXAXIS,LYAXIS
*      VARIABLES FOR COMMON BLOCK PRTCTL
       INTEGER PRTCNT,PROFLS
       REAL TOFTEN,DOFTEN,NXTPRT

       COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
       SAVE /GLOBAL/
       COMMON /PLT/ X,LXAXIS,LYAXIS
       SAVE /PLT/
       COMMON /PRTCTL/ TOFTEN,DOFTEN,PROFLS,PRTCNT,NXTPRT
       SAVE /PRTCTL/

       DATA MSGANG/20.0/

            DO 100,I=1,PTS
*              .001 IS AN ADJUSTMENT BECAUSE OF EXTRA HEIGHT ADDED
*              IN MAKING AN EXPONENTIAL BOTTOM.
               Y(I)=UNTHGT*(H(I) - 0.001)
100         CONTINUE

            CALL CURVE(X,Y,1001,0)
*           COMPUTE X,Y DISTANCE FROM ORIGIN OF YEAR-OF-PROFILE MESSAGE
            YEARX = DSTINC * (START-1) * LXAXIS / LENGTH
            YEARY = (SEALVL + 0.01) * LYAXIS
            CALL ANGLE(MSGANG)
            CALL REALNO(TIME*TIMCOR,1,YEARX,YEARY)
            CALL MESSAG('YEARS$',100,'ABUT','ABUT')
*           CHANGE-OF-PARAMETERS MESSAGE IF APPROPRIATE
            IF (PRTCNT .EQ. PROFLS) THEN
               CALL MESSAG(' END PARM SET$',100,'ABUT','ABUT')
            ENDIF

99999       CONTINUE

            RETURN
            END

***************************************************************************

*    SUBROUTINE GETPMS
*    COLLECTS FROM FILE USER THE PARAMETERS THAT MAY BE CHANGED DURING THE RUN
```

```
*   COMPUTES CONSTANTS THAT CHANGE WITH EACH SET OF PARMS
*
      SUBROUTINE GETPMS

      REAL YEARS,QTDST,PRTDST,QB,QBTHIN,PLUME,PORQB,DG2RAD,YR2SEC
      REAL DAY2SC,XINC

*     VARIABLES FOR COMMON BLOCK,GLOBAL 6
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL DIFSN,TOL
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      LOGICAL TIMCTL,QUIT
*     VARIABLES FOR COMMON BLOCK GRAIN
      REAL CNTN(4), LMDA(4), DENSTY(4), DEPTH
*     VARIABLES FOR COMMON BLOCK PRTCTL
      INTEGER PRTCNT,PROFLS
      REAL TOFTEN,DOFTEN,NXTPRT
*     VARIABLES FOR COMMON BLOCK,  FINDVX
      REAL CHANNL,V0,XB
*     VARIABLES FOR COMMON BLOCK, BYIN
      REAL RUNOUT,QBYPAS
      INTEGER  RUNNUM
*     VARIABLES FOR COMMON BLOCK, ITRCST
      REAL TSTEP,YRCOR,DMPALL,QDUMP,A,B,D
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/
      COMMON /GRAIN/ CNTN,LMDA,DENSTY,DEPTH
      SAVE /GRAIN/
      COMMON /PRTCTL/ TOFTEN,DOFTEN,PROFLS,PRTCNT,NXTPRT
      SAVE /PRTCTL/
      COMMON /FINDVX/ CHANNL,V0,XB
      SAVE /FINDVX/
      COMMON /BYIN/ RUNOUT,RUNNUM,QBYPAS
      SAVE /BYIN/
      COMMON /ITRCST/ TSTEP, YRCOR,DMPALL,QDUMP,A,B,D
      SAVE /ITRCST/
      DATA DMPLGT,HEAVY/200.0,1750.0/

       DG2RAD = 3.141582654/180.0
       YR2SEC = 3600.0 * 24.0 * 365.0
       DAY2SC = 3600.0 * 24.0
       IF (TIMCTL) THEN
*        CONTROL BY TIME PASSED
         READ(11,*,END=9900,ERR=9901)YEARS
         READ(11,*,END=9900,ERR=9901)TOFTEN
         PROFLS = INT(0.5 + YEARS/TOFTEN)
         YEARS = PROFLS*TOFTEN
         WRITE(20,'(''RUN FOR'',F8.1,'' YEARS'')') YEARS
         WRITE(20,'(''PRINT EVERY'',F8.1,'' YEARS'')')TOFTEN
C        CONVERT TO DIMENSIONLESS TIME
         TOFTEN = TOFTEN/TIMCOR
       ELSE
*        CONTROL BY DISTANCE PROGRADED
         READ(11,*,END=9900,ERR=9901)QTDST
         READ(11,*,END=9900,ERR=9901)PRTDST
C        CONVERT TO NUMBER OF POINTS IN THE VECTOR
         DOFTEN = (0.5 + PRTDST * 1000.0/DSTINC)
         PRTDST = DOFTEN * DSTINC / 1000.0
         PROFLS = INT(0.5 + QTDST/PRTDST)
         QTDST = (PROFLS-1) * PRTDST
         WRITE(20,'(''PROGADE'',F8.2,'' KM'')')QTDST
         WRITE(20,'(''PRINT EVERY'',F8.2,'' KM'')')PRTDST
       ENDIF
       READ(11,*,END=9900,ERR=9901)QB
       WRITE(20,'(''QB: '',F7.1,'' KG/SEC'')')QB
       QBTHIN = QB * YR2SEC / (WIDTH * HEAVY)
       READ(11,*,END=9900,ERR=9901)PORQB
       WRITE(20,'(''PORTION QB IN BYPASS'',F5.3)')PORQB
       READ(11,*,END=9900,ERR=9901)RUNOUT
       WRITE(20,'(''RUNOUT DISTANCE'',F6.2,'' KM'')')RUNOUT
```

```
C      CONVERT TO NUMBER OF POINTS
       RUNNUM = INT(0.5 + RUNOUT*1000.0/DSTINC)
C      CONVERT RUNOUT TO METRES
       RUNOUT = RUNOUT * 1000.0
*      GET THE PARAMETERS FOR DEPOSITING HEMIPHELAGIC FLUX
*      EACH OF THE NEXT FOUR LINES IN THE FILE SHOULD BE FORMATTED
*      INITIAL CONCENTRATIN,LMDA,DENSITYCORRECTION.
       DO 105,I=1,4
           READ(11,*,END=9900,ERR=9901)CNTN(I),LMDA(I),DENSTY(I)
*          CONVERT LMDA TO 1/SEC
           LMDA(I) = LMDA(I)/DAY2SC
*          CONVERT CONCENTRATION TO KG/CUBIC METRE
           CNTN(I) = CNTN(I)/1000.0
105    CONTINUE
       READ(11,*,END=9900,ERR=9901)V0
       WRITE(20,'(''VELOCITY '',F6.1,''M/SEC'')')V0
       READ(11,*,END=9900,ERR=9901)CHANNL
       WRITE(20,'(''CHANNEL WIDTH '',F6.1,''M'')')CHANNL
       READ(11,*,END=9900,ERR=9901)DEPTH
       WRITE(20,'(''DEPTH OF CHANNEL'',F7.2,'' M'')')DEPTH
       READ(11,*,END=9900,ERR=9901)PLUME
       WRITE(20,'(''MAX PLUME WIDTH '',F6.1,'' KM'')')PLUME
*      CONVERT TO METRES
       PLUME = PLUME * 1000.0
*      CALCULATE THE DISTANCE, XB, FROM THE RIVER MOUTH WHERE THE
*      PLUME REACHES MAXIMUM WIDTH.
       XB = 0.5 * (PLUME - CHANNL)/ TAN(12.0 * DG2RAD)

        XINC = 1.0 / (PTS -1)
        IF (DIFSN .LT. 1.0) THEN
          TSTEP = .208/TIMCOR
        ELSE
          TSTEP = .999*XINC*XINC/(2.0*DIFSN)
        ENDIF
        A = TSTEP*DIFSN/(XINC*XINC)
        B = 1.0 - 2.0*A
        D = TSTEP*LENGTH*LENGTH/UNTHGT
        YRCOR = UNTHGT/(TSTEP*TIMCOR)
        QBYPAS = D*QBTHIN*PORQB
        DMPALL = D * QBTHIN / DMPLGT
        QDUMP = D*QBTHIN*(1.0-PORQB)/DMPLGT
        CALL SEDFLX(D)

       GOTO 99999
9900   WRITE(20,'(''NO RECORD IN USER'')')
9901   WRITE(20,'(''ERROR READING USER'')')
99999  CONTINUE
       RETURN
       END

****************************************************************************

*      SUBROUTINE SETST   ---   ESTABLISHES INDEX OF THE POINT
*      THAT REPRESENTS THE RIVER MOUTH
*
      SUBROUTINE SETST

*       VARIABLES FOR COMMON BLOCK,GLOBAL
        INTEGER START,PTS,ENDDMP,LOWPT
        REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
        REAL DIFSN,TOL
        LOGICAL TIMCTL,QUIT
*       VARIABLES FOR COMMON BLOCK, FLXDWN
        REAL FLXTRM(2001)
        COMMON /FLXDWN/ FLXTRM
        SAVE /FLXDWN/
        COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C      LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
        SAVE /GLOBAL/

        START = 1
```

```
        IF (H(2) .GT. (SEALVL-TOL)) THEN
*         WHILE H(START+1) >= SEALVL, MOVE START TOWARD THE SEA
100       CONTINUE
            IF (H(START+1) .LT. (1.0-TOL)*SEALVL) GOTO 200
            START = START + 1
            GOTO 100
200       CONTINUE
*         MOVE THE FLX VECTOR SO WE START DEPOSITING SUSPENDED SEDIMENT
*         JUST IN FRONT OF THE RIVER MOUTH.
          DO 300 I=START+PTS-1,START,-1
             FLXTRM(I) = FLXTRM(I-START+1)
300       CONTINUE
          DO 400,I=1,START-1
             FLXTRM(I) = 0.0
400       CONTINUE
        ENDIF

        RETURN
        END

*****************************************************************************

*     SUBROUTINE FSTPRT
*          IF THE PROFILES ARE ARE OUTPUTTED AFTER A TIME INTERVAL,
*          CALCULATE THE TIME OF THE FIRST OUTPUT.
*          IF THEY ARE OUTPUTTED AFTER THE MODEL HAS PROGRADED A
*          GIVEN DISTANCE, CALCULATE THE INDEX OF THE POINT THAT
*          WILL REPRESENT THE RIVER MOUTH DURING THE FIRST OUTPUT.

      SUBROUTINE FSTPRT

*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,ENDDMP,LOWPT
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
      LOGICAL TIMCTL,QUIT
*     VARIABLES FOR COMMON BLOCK PRTCTL
      INTEGER PRTCNT,PROFLS
      REAL TOFTEN,DOFTEN,NXTPRT
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      COMMON /PRTCTL/ TOFTEN,DOFTEN,PROFLS,PRTCNT,NXTPRT
      SAVE /PRTCTL/

      IF(TIMCTL) THEN
        NXTPRT = TIME + TOFTEN
      ELSE
        NXTPRT = (START + DOFTEN) *1.0
      ENDIF

      RETURN
      END

*****************************************************************************

*      SUBROUTINE PROGRD -- MOVES THE RIVER MOUTH FORWARD BY INCREMENTING
*      START. SHIFTS THE VALUES IN THE ARRAY FLXTRM TO THE
*      "RIGHT".  IF THE STORAGE OF PROFILES IS CONTROLLED BY THE DISTANCE
*      PROGRADED, A CHECK IS MADE TO DETERMINE IF IT IS TIME FOR A PROFILE.

      SUBROUTINE PROGRD(PRINT)

      INTEGER OLDST,DIFF
      LOGICAL PRINT
*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,LOWPT,ENDDMP
      REAL DIFSN,TOL
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      LOGICAL TIMCTL,QUIT
*     VARIABLES FOR COMMON BLOCK PRTCTL
      INTEGER PRTCNT,PROFLS
```

```
      REAL TOFTEN,DOFTEN,NXTPRT
*     VARIABLES FOR COMMON BLOCK, FLXDWN
      REAL FLXTRM(2001)
      COMMON /PRTCTL/ TOFTEN,DOFTEN,PROFLS,PRTCNT,NXTPRT
      SAVE /PRTCTL/
      COMMON /FLXDWN/ FLXTRM
      SAVE /FLXDWN/
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/

      OLDST = START
      IF(H(LOWPT) .GT. H(START))THEN
        LOWPT = START
      ENDIF
      START = START + 1
*     WHILE H(START+1) IS ABOVE SEA LEVEL AND WE HAVEN'T MOVED BEYOND THE
*     BOUNDARIES OF THE SYSTEM, CONTINUE TO PROGRADE
50    CONTINUE
         IF(START.GE.PTS)GOTO 60
         IF(H(START+1) .LT. (1.0-TOL)*SEALVL)GOTO 60
*        IF THE POINT WE'RE LEAVING BEHIND IS LOWER THAN THE LOWEST
*        POINT ON THE TOPSET, CHANGE THE STORED INDEX.
         IF(H(LOWPT) .GT. H(START))THEN
           LOWPT = START
         ENDIF
         START = START+1
         GOTO 50
60    CONTINUE

      IF(START .LT. PTS)THEN
*         ADJUST THE FLX VECTOR SO WE'RE DEPOSITING JUST BEYOND THE
*         RIVER MOUTH
          DIFF = START - OLDST
          DO 70,I=START+PTS-1,START,-1
             FLXTRM(I) = FLXTRM(I-DIFF)
70       CONTINUE

         DO 80,I=OLDST,START-1
            FLXTRM(I) =0.0
80      CONTINUE
*           WRITE THE CURRENT TIME TO FILE'ADVANCE'
            WRITE(13,'(F9.2,''A'')')TIME*TIMCOR

      ELSE
*           PRINT A WARNING AND QUIT
            QUIT = .TRUE.
            WRITE(20,'(''TRYING TO PROGRADE BEYOND SILL'')')
      ENDIF

            RETURN
            END

*********************************************************************


*********************************************************************
*      SUBROUTINE REGRAD -- MOVES THE RIVER MOUTH BACKWARD BY DECREMENTING
*      START. SHIFTS THE VALUES IN THE ARRAY FLXTRM  TO THE
*      "LEFT".

      SUBROUTINE REGRAD

      INTEGER OLDST,DIFF
*     VARIABLES FOR COMMON BLOCK,GLOBAL
      INTEGER START,PTS,LOWPT,ENDDMP
      REAL DIFSN,TOL
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      LOGICAL TIMCTL,QUIT
```

```
*     VARIABLES FOR COMMON BLOCK, FLXDWN
      REAL FLXTRM(2001)
      COMMON /FLXDWN/ FLXTRM
      SAVE /FLXDWN/
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/



      OLDST = START
      IF(H(LOWPT).LE.H(START))THEN
        START = LOWPT
      ENDIF
      START = START-1
*     WHILE H(START) IS BELOW SEA LEVEL AND WE HAVEN'T MOVED BEYOND THE
*     BOUNDARIES OF THE SYSTEM, CONTINUE TO PROGRADE
50    CONTINUE
         IF(START.EQ.0)GOTO 60
         IF(H(START) .GE. (1-TOL)*SEALVL)GOTO 60
         START = START-1
         GOTO 50
60    CONTINUE

      IF(START .GE. 1)THEN
*         ADJUST THE FLX VECTOR SO WE'RE DEPOSITING JUST BEYOND THE
*         RIVER MOUTH
          DIFF = OLDST - START
          DO 70,I=START,START+PTS-1
             FLXTRM(I) = FLXTRM(I+DIFF)
70       CONTINUE

         DO 80,I=START+PTS,OLDST+PTS-1
            FLXTRM(I) =0.0
80      CONTINUE
*       IF WE'RE REGRADING TO/PAST THE LOWEST POINT ON THE TOPSET, LOCATE THE
*       NEW LOWEST POINT
        IF(LOWPT.GE.START)THEN
           LOWPT = 1
           I=2
85         CONTINUE
              IF(I .GE. START)GOTO 90
              IF(H(I) .LT. H(LOWPT))THEN
                LOWPT = I
              ENDIF
              I=I+1
              GOTO 85
90         CONTINUE
        ENDIF

*           WRITE THE CURRENT TIME TO FILE'ADVANCE'
            WRITE(13,'(F9.2,''B'')')TIME*TIMCOR



      ELSE
            QUIT = .TRUE.
            WRITE(20,'(''TRYING TO REGRADE BEYOND ORIGIN'')')
      ENDIF

            RETURN
            END
*******************************************************************************



*******************************************************************************

*     SUBROUTINE CLENUP
```

```
*     CALCULATES PROGRADATION RATES AT TOFT YEAR INTERVALS
*     MAKES THE INFORMATION IN TRACK PRESENTABLE
      SUBROUTINE CLENUP

      REAL OLDTIM,NXTTIM,RATE,TOFT,OUTTIM,TRACKX,SLOPEX,OUTSLP,PTSLP
      REAL OUTSRC,OUTSED,OUTTOT,NEWTIM
       CHARACTER DRECTN
*     VARIABLES FROM COMMON BLOCK GLOBAL
      REAL LENGTH,TIMCOR,TIME,SEALVL,UNTHGT,H(1001),DSTINC,WIDTH
      REAL DIFSN,TOL
      INTEGER START,PTS,ENDDMP,LOWPT
      LOGICAL TIMCTL,QUIT
      COMMON /GLOBAL/ TIMCOR,TIME,SEALVL,UNTHGT,H,START,PTS,DSTINC,
     C    LENGTH,WIDTH,TIMCTL,QUIT,DIFSN,ENDDMP,TOL,LOWPT
      SAVE /GLOBAL/

      DATA OLDTIM,TOFT/0.0,200.0/

      ENDFILE 20
      WRITE(20,'('' PROGRADATION RATE AS A FUNCTION OF TIME'')')
      WRITE(20,'(''   TIME        PROGRADATION RATE'')')
      WRITE(20,'(''  (YEARS)           (M/YR)   '')')
      REWIND(13)
      NXTTIM = TOFT
      READ(13,'(F9.2,A)',END=90,ERR=95)NEWTIM,DRECTN

*     WHILE NOT END-OF-FILE, READ AND CALCULATE PROGADATION RATE
10    CONTINUE
        IF(NEWTIM .GE. NXTTIM)THEN
          NXTTIM = NXTTIM + TOFT
          RATE = DSTINC/(NEWTIM - OLDTIM)
          IF(DRECTN .EQ. 'B')THEN
*          WE'VE BEEN REGRADING
            RATE = -RATE
          ENDIF
          OUTTIM = (NEWTIM + OLDTIM) / 2.0
          WRITE(20,'(F8.0,9X,F8.2)')OUTTIM,RATE
        ENDIF
        OLDTIM = NEWTIM
        READ(13,'(F9.2,A)',END=99,ERR=95)NEWTIM,DRECTN
      GOTO 10

90    WRITE(20,'(''SUBROUTINE,CLENUP--NO RECORDS IN ADVANCE'')')
      GOTO 99
95    WRITE(20,'(''SUBROUTINE,CLENUP--ERROR READING ADVANCE'')')
99    CONTINUE
      ENDFILE 20

*     CLEAN UP THE TRACKING OUTPUT. FIRST OUTPUT ACCUMULATION
*     RATE AND SLOPE TRACKX KM'S FROM ORIGIN.
      REWIND 15
      READ(15,'(2F6.1)')TRACKX,SLOPEX
      WRITE(20,'(''ACCUMULATION RATES AND SLOPE'',F6.2,
     C     '' KM FROM ORIGIN'')')TRACKX
      WRITE(20,'('' '')')
      WRITE(20,'('' TIME'',4X,''SLOPE'',4X,''HEMIPELAGIC '',
     C     ''HEMIPELAGIC '',''TOTAL'')')
      WRITE(20,'('' (YEARS) (DEGREE) RAIN'',8X,''RAIN AND'',
     C     4X,''ACCUMULATION'')')
      WRITE(20,'(18X,''(CM/A)'',6X,''BYPASS'',6X,''(CM/A)'')')
      WRITE(20,'(30X,''(C/A)'')')
      READ(15,'(F12.1,5F12.8)',END=900,ERR=950)OUTTIM,OUTSLP,PTSLP,
     C    OUTSED,OUTSRC,OUTTOT

100   CONTINUE
        IF(OUTSED .EQ. -95)GOTO 200
        WRITE(20,'(F8.0,F6.1,F8.1,2F12.1)')OUTTIM,PTSLP,OUTSED,
     C     OUTSRC,OUTTOT
        READ(15,'(F12.1,5F12.8)',END=200,ERR=950)OUTTIM,OUTSLP,PTSLP,
     C    OUTSED,OUTSRC,OUTTOT
```

```
          GOTO 100
200     CONTINUE
        ENDFILE 20

        GOTO 999
900     WRITE(20,'(''SUBROUTINE,CLENUP—NO RECORDS IN TRACK'')')
        GOTO 999
950     WRITE(20,'(''SUBROUTINE,CLENUP—ERROR READING TRACK'')')
999     CONTINUE


*       CLEAN UP THE TRACKING OUTPUT. NEXT OUTPUT SLOPE
*       SLOPE SLOPEX KMS FROM RIVER MOUTH.
        REWIND 15
        READ(15,'(2F6.1)')TRACKX,SLOPEX
        WRITE(20,'(''SLOPE'',F6.2,
       C      '' KM FROM RIVER MOUTH AT TIME INTERVALS'')')SLOPEX
        WRITE(20,'('' '')')
        WRITE(20,'('' TIME'',4X,''SLOPE'')')
        WRITE(20,'('' (YEARS) (DEGREES)'')')
        READ(15,'(F12.1,5F12.8)',END=9000,ERR=9500)OUTTIM,OUTSLP,PTSLP,
       C    OUTSED,OUTSRC,OUTTOT

1000    CONTINUE
          IF(OUTSLP .EQ. -95)GOTO 2000
          WRITE(20,'(F8.0,F6.1)')OUTTIM,OUTSLP
          READ(15,'(F12.1,5F12.8)',END=2000,ERR=9500)OUTTIM,OUTSLP,PTSLP,
       C    OUTSED,OUTSRC,OUTTOT
          GOTO 1000
2000    CONTINUE

        GOTO 9999
9000    WRITE(20,'(''SUBROUTINE,CLENUP—NO RECORDS IN TRACK'')')
        GOTO 9999
9500    WRITE(20,'(''SUBROUTINE,CLENUP—ERROR READING TRACK'')')
9999    CONTINUE
        RETURN
        END
```

```
      PROGRAM GRAIN
*
*    AUTHOR: E.A. CALABRESE
*            ATLANTIC GEOSCIENCE CENTRE
*            GEOLOGICAL SURVEY OF CANADA
*            BOX 1006, DARTHMOUTH, N.S.
*            CAN.   B2Y 4A2
*************************************************************************
*
*   BACKGROUND
*       GRAIN MODELS THE DEPOSITION OF SEDIMENT IN A TWO DIMENSIONAL PLUME
*       WHICH SPREADS AFTER ENTERING A STRATIFIED BASIN TO SOME
*       MAXIMUM WIDTH DETERMINED BY THE WIDTH OF THE BASIN.
*       GRAIN SIZES ARE DEPOSITED SEPARATELY.
*       THE PROGRAM COMPUTES THE PROPORTION OF THE DEPOSIT MADE UP
*       OF EACH GRAIN SIZE AS A FUNTION OF DISTANCE FROM
*       THE RIVER MOUTH. ALSO USING OUTPUT FROM THE MODEL, DELTA,
*       COMPUTES GRAIN SIZE FRACTIONATION AT A PARTICULAR POINT IN THE
*       DELTA AS A FUNCTION OF TIME.
****************************************************************************
*
*  FILES
*      USER -- PROVIDES THE PARAMETERS REQUIRED TO RUN THE MODEL.
*
*      ADVANCE -- OUTPUT FROM DELTA, A LIST OF TIMES AT WHICH THE
*                 MODEL WAS PROGRADED
*
*      DEPOSIT -- OUTPUT FROM GRAIN, GRAIN SIZE FRACTIONATION AS A
*                 FUNCTION OF DISTANCE AND TIME
*
****************************************************************************
*
*     VARIABLES
*

*   BINS -- NUMBER OF COLLECTION BINS ALONG THE LENGTH OF THE
*           DELTA (PTS-1)
*   CHANNL --  CHANNEL WIDTH. WIDTH OF THE RIVER MOUTH IN METRES.
*   CNTN -- CONCENTRATION. A VECTOR CONTAINING THE CONCENTRATION
*           OF SUSPENDED SEDIMENT AT THE RIVER MOUTH. BROKEN DOWN BY
*           GRAIN SIZE

*   DAY2SC -- NUMBER OF SECONDS IN A DAY
*   DEPTH-- DEPTH OF CHANNEL AT RIVER MOUTH
*   DG2RAD -- FACTOR TO CONVERT DEGREES TO RADIANS(PI/180)
*   DSTINC -- UNSCALED DISTANCE BETWEEN COLLECTION POINTS
*             (METRES)

*   E -- 2.71828182846...

*   FLX -- VECTOR STORING THE AVERAGE SUSPENDED SEDIMENT
*        PER YEAR AT EACH COLLECTION POINT ALONG THE
*           PROFILE

*   GRAINS -- NUMBER OF GRAIN SIZE FRACTIONS BEING CONSIDERED

*   I -- USED AS AN INDEX
*   I0 -- MASS OF SEDIMENT IN A COLUMN OF WATER AT THE RIVER MOUTH

*   J -- USED AS AN INDEX

*   K -- POWER OF E IN CALCULATING THE X-COMPONENT OF
*        VELOCITY.

*   LMDA --  (LAMDA) REMOVAL RATE CONSTANT FOR HEMIPELAGIC RAIN
*            READ IN UNITS 1/DAY. CONVERTED IMMEDIATELY TO
*            1/SEC
*   LENGTH -- DISTANCE FROM MODEL ORIGIN TO SILL. READ IN IN KM.
*             CONVERTED IMMEDIATELY TO METRES.
```

```
*     MASS -- MASS OF SEDIMENT DEPOSITED(AN ACCUMULATOR)

*   PLUME -- FINAL WIDTH OF RIVER PLUME(WHEN NO LONGER SPREADING)
*   PNTFLX --  VECTOR OF COLLECTION POINTS IN A CROSS-SECTION OF
*            THE DELTA PERPENDICULAR TO FJORD WALLS. USED IN CALCULATING
*            SUSPENDED SEDIMENT DEPOSITION RATE
*
*   PORTN -- A VECTOR WHICH STORES THE PROPORTION OF THE SEDIMENT IN
*            EACH GRAIN SIZE
*   PTS -- NUMBER OF POINTS AT WHICH SEDIMENTATION IS BEING CALCULATED
*   PTSACC -- THE NUMBER OF COLLECTION POINTS IN A CROSS-SECTION
*            FROM THE ORIGIN

*   QSDOWN -- AMOUNT OF SUSPENDED SEDIMENT DEPOSITED
*   ROW -- AN INDEX (A ROW OF POINTS FORMS A CROSS-SECTION)

*   T -- (TIME) A VECTOR USED FOR ACCUMULATING THE TIME REQUIRED
*        FOR EACH COLUMN OF WATER TO REACH THE CURRENT DISTANCE
*        FROM THE FJORD HEAD. USED TO CALCULATE FLUX
*   TIMNEW -- (NEW TIME) TIME OF PROGRADATION NEWLY READ FROM ADVANCE
*             CONTROL BY DISTANCE)
*   TIMNXT -- (NEXT TIME) THE NEXT TIME(IN YEARS) AT WHICH TO
*             CALCULATE PROGRADATION RATE
*   TIMOLD -- (OLD TIME) THE LAST TIME READ FROM ADVANCE
*   TINC -- TIME REQUIRED FOR A COLUMN OF WATER TO ADVANCE ACROSS
*           ONE SQUARE IN THE GRID
*   TOFT -- TIME(IN YEARS) BETWEEN CALCULATION OF PROGRADATION RATES
*   TOL -- (TOLERANCE) USED TO ALLOW FOR ROUND-OFF ERROR
*   TRACKX -- PERIODICALLY THE SLOPE AND ACCUMULATION RATES
*             A GIVEN DISTANCE FROM THE ORIGIN ARE RECORDED.
*             THIS DISTANCE IS TRACKX.
*   TRKPNT -- INDEX OF THE COLLECTION POINT THAT REPRESENTS TRACKX
*   V0 -- VELOCITY OF RIVER WATER AT RIVER MOUTH(M/SEC)
*   VX -- VECTOR, X-COMPONENTS OF VELOCITY AT THE COLLECTION PTS
*         IN THE CROSS-SECTION PERPENDICLAR TO FJORD WALLS.
*         USED IN COMPUTING HEMIPELAGIC FLUX
*   VXBACK -- THE X-COMPONENT OF VELOCITY ONE SQUARE CLOSER TO THE
*             RIVER MOUTH AND HALF A SQUARE FARTHER FROM THE CENTRE
*             OF THE PLUME THAN THE CURRENT POINT.
*   VXFRD -- THE X-COMPONENT OF VELOCITY ONE SQUARE FARTHER FROM THE
*            RIVER MOUTH AND HALF A SQUARE FARTHER FROM THE CENTRE
*            OF THE PLUME THAN THE CURRENT POINT.
*   VY -- Y-COMPONENT(LATERAL COMPONENT) OF VELOCITY IN THE RIVER PLUME
*   VYINC -- INCREASE IN THE Y-COMPONENT OF VELOCITY AS WE MOVE OUTWARD
*          ONE SQUARE FROM THE CENTRE OF THE RIVER PLUME.


*   X -- VECTOR CONTAINING THE DISTANCE OF EACH OF THE COLLECTION
*        POINTS FROM THE RIVER MOUTH. USED IN PLOTTING
*   X0 -- THE DISTANCE FROM THE RIVER MOUTH WHERE PLUME DYNAMICS
*         CHANGE FROM ZONE OF ESTABLISHMENT TO ZONE OF ESTABLISHED
*         FLOW
*   XB -- DISTANCE FROM RIVER MOUTH WHERE PLUME ACHIEVES
*         MAXIMUM WIDTH
*   XBACK -- X-COORDINATE OF THE POINT IN THE GRID ONE SQUARE CLOSER
*            TO THE RIVER MOUTH THAN CURRENT POINT
*   XBPNT -- THE INDEX OF THE COLLECTION POINT THAT REPRESENTS THE
*            THE POINT IN THE DELTA WHERE THE PLUME REACHES MAXIMUM
*            WIDTH
*   XFRD -- X-COORDINATE OF THE POINT IN THE GRID ONE SQUARE FARTHER
*           FROM THE RIVER MOUTH THAN CURRENT POINT
*   XPOSTN -- DISTANCE FROM THE RIVER MOUTH TO CURRENT POSITION
*             (X POSITION). GIVEN IN METRES.

*   Y -- VECTOR OF POINTS REPRESENTING LATERAL POSITION IN THE RIVER.
*        USED IN CALCULATING HEMIPELAGIC FLUX.
      REAL FLX(1001,4),DSTINC,LENGTH
      INTEGER I,J,GRAINS,PTS
      COMMON /MATRIX/ FLX,GRAINS,DSTINC
      DATA GRAINS,PTS,BINS/4,1001,1000/
```

```
      OPEN(10,FILE='DEPOSIT',STATUS='NEW')
      OPEN(11,FILE='USER',STATUS='OLD')
      OPEN(13,FILE='ADVANCE',STATUS='OLD')

*     READ PAST FIRST TWO RECORDS IN USER TO COLLECT LENGTH AND
*     HENCE COMPUTE DSTINC
      READ(11,*,END=9900,ERR=9901)
      READ(11,*,END=9900,ERR=9901)
      READ(11,*,END=9900,ERR=9901)LENGTH
      DSTINC = LENGTH*1000.0/(1.0*(PTS-1))

*     MODEL THE FLUX RATE AT EACH OF THE COLLECTION POINTS.
      CALL SEDFLX(PTS)

*       COMPUTE AND OUTPUT GRAIN SIZE FRACTIONATION AS A FUNCTION OF DISTANCE
        CALL DSTNCE(PTS)

*       COMPUTE AND OUTPUT GRAIN SIZE FRACTIONATION AS A FUNCTION OF TIME
*       USING OUTPUT FROM DELTA
        CALL TIMINT(PTS)

      GOTO 99999
9900  WRITE(*,'(''PGM GRAIN — NO RECORD IN USER'')')
      GOTO 99999
9901  WRITE(*,'(''PGM GRAIN — ERROR READING USER'')')
99999 CLOSE(10)
      CLOSE(11)
      CLOSE(13)
      STOP
      END

**********************************************************************

*     SUBROUTINE DSTNCE  — CALCULATES THE GRAIN SIZE FRACTIONATION
*                        IN HEMIPELAGIC RAIN AS DISTANCE FROM THE
*                        RIVER MOUTH INCREASES
      SUBROUTINE DSTNCE(PTS)
      REAL FLX(1001,4),MASS,PORTN(4),DSTINC
      INTEGER I,J,GRAINS,PTS
      COMMON /MATRIX/ FLX,GRAINS,DSTINC
      SAVE /MATRIX/

*     WRITE HEADINGS
      WRITE(10,'(''PARTICLE SIZE FRACTIONATION WITH DISTANCE '',
     C     ''FROM THE RIVER MOUTH'')')
      WRITE(10,'(''DISTANCE  CS      MS      FS      CLAY'')')
      WRITE(10,'(''   (KM)   (%)    (%)    (%)    (%)   '')')
*     UNTIL WE'VE RUN OUT OF POINTS, COMPUTE THE PARTICLE SIZE
*     FRACTIONATION AT INTERVALS. THE INTERVAL IS SMALLER
*     CLOSE TO THE RIVER MOUTH
      I = 1
100   CONTINUE
         MASS = 0.0
*        CALCULATE THE MASS OF SEDIMENT DEPOSITED PER UNIT AREA
         DO 50 J=1,GRAINS
              MASS = MASS + FLX(I,J)
50       CONTINUE

*        CALCULATE THE FRACTION OF EACH GRAIN SIZE
         DO 60 J=1,GRAINS
           PORTN(J) = 100.0*FLX(I,J)/MASS
60       CONTINUE

         WRITE(10,'(F7.1,F6.0,3F7.0)')(I-1)*DSTINC/1000.0,
     C        (PORTN(J),J=1,GRAINS)

         IF(I .LT. 100)THEN
           I = I + 10
         ELSE
           I = I + 50
         ENDIF
```

```
      IF(I .LE. PTS) GOTO 100

      RETURN
      END
********************************************************************

*     SUBROUTINE TIMINT  — CALCULATES THE GRAIN SIZE FRACTIONATION
*                             WITH CHANGING TIME
*                        IN HEMIPELAGIC RAIN AT A GIVEN DISTANCE
*                        FROM THE ORIGINAL RIVER MOUTH
      SUBROUTINE TIMINT(PTS)
      REAL FLX(1001,4),TRACKX,MASS,PORTN(4),TIMNXT,TINC,TIMNEW
      REAL TIMOLD,TOL
      INTEGER I,GRAINS,TRKPNT,PTS
      LOGICAL EOF
      CHARACTER DRECTN
      COMMON /MATRIX/ FLX,GRAINS,DSTINC
      SAVE /MATRIX/
      DATA TOL,TINC/0.000001,200.0/
      DATA EOF/.FALSE./

*     GET THE DISTANCE(TRACKX) OF THE POINT WE'RE INTERESTED IN FROM
*     THE FILE WITH THE PARMS IN IT
      REWIND 11
*     READ PAST OTHER RECORDS
      DO 100,I=1,6
        READ(11,*,END=9900,ERR=9901)
100   CONTINUE
      READ(11,*,END=9900,ERR=9901)TRACKX
      TRKPNT = INT(TRACKX*1000.0/DSTINC) + 1

      ENDFILE 10
*     WRITE HEADINGS
      WRITE(10,'(''PARTICLE SIZE FRACTIONATION AT'',F6.1,'' KM'')')
     C      TRACKX
      WRITE(10,'('' '')')
      WRITE(10,'(''  TIME   CS    MS    FS    CLAY'')')
      WRITE(10,'(''  (YRS)  (%)   (%)   (%)   (%) '')')

*     OUTPUT THE INITIAL FRACTIONATION
        MASS = 0.0
        DO 30 I=1,GRAINS
             MASS = MASS + FLX(TRKPNT,I)
30      CONTINUE

*       COMPUTE THE PORTION OF EACH GRAIN SIZE IN THE SEDIMENT
        DO 40 I=1,GRAINS
          PORTN(I) = 100.0*FLX(TRKPNT,I)/MASS
40      CONTINUE

        WRITE(10,'(F7.0,F5.0,3F6.0)')0.0,(PORTN(I),I=1,GRAINS)

*     INITIALIZE
      TIMOLD = 0.0
      TIMNXT = TINC
      READ(13,'(F9.2,A)',END=900,ERR=9902)TIMNEW,DRECTN


*     WHILE TRACKPT IS BETWEEN THE RIVER MOUTH(>1) AND THE SILL(<PTS)
*     AND WE HAVEN'T REACHED THE END OF ADVANCE, COMPUTE AND WRITE
*     THE SEDIMENTATION RATE AT THIS TIME, AND THEN FIND
*     THE POSITION OF TRACKX(RELATIVE TO RIVER MOUTH)
*     AT THE TIME THE NEXT OUTPUT IS REQUIRED
200   CONTINUE
        IF((TRKPNT .LE. 1) .OR. (TRKPNT .GE. PTS))GOTO 900
        IF(TIMNXT .LE. (TIMNEW+TOL))THEN
*         COMPUTE THE MASS OF SEDIMENT DEPOSITED HERE PER YEAR
          MASS = 0.0
          DO 300 I=1,GRAINS
             MASS = MASS + FLX(TRKPNT,I)
```

```
300       CONTINUE

*         COMPUTE THE PORTION OF EACH GRAIN SIZE IN THE SEDIMENT
          DO 400 I=1,GRAINS
            PORTN(I) = 100.0*FLX(TRKPNT,I)/MASS
400       CONTINUE

*         PRINT THESE RESULTS UNTIL THE NEXT OUTPUT TIME(TIMNXT) IS
*         AFTER THE TIME OF THE NEXT PROGRADATION.
600       CONTINUE
            WRITE(10,'(F7.0,F5.0,3F6.0)')TIMNXT,(PORTN(I),I=1,GRAINS)
            TIMNXT = TIMNXT + TINC
          IF(TIMNXT .LE. (TIMNEW+TOL))GOTO 600

        ENDIF

*       GET THE TIME OF THE NEXT PROGRADATION (REGRESSION)
*       MOVE TRKPNT ONE STEP
*       CLOSER TO(AWAY FROM) THE RIVER MOUTH
        TIMOLD = TIMNEW
        READ(13,'(F9.2,A)',END=900,ERR=9902)TIMNEW,DRECTN
        IF (DRECTN .EQ. 'B') THEN
           TRKPNT = TRKPNT + 1
        ELSE
           TRKPNT = TRKPNT - 1
        ENDIF

        GOTO 200
900   CONTINUE

      GOTO 99999

9900  WRITE(*,'(''SUBROUTINE TIMINT -- NO RECORD IN USER'')')
      GOTO 99999
9901  WRITE(*,'(''SUBROUTINE TIMINT -- ERROR READING USER'')')
      GOTO 99999
9902  WRITE(*,'(''SUBROUTINE TIMINT -- ERROR READING ADVANCE'')')
      GOTO 99999
9903  WRITE(*,'(''SUBROUTINE TIMINT -- NO RECORD IN ADVANCE'')')
99999 CONTINUE
      RETURN
      END
******************************************************************
*       SUBROUTINE,SEDFLX
*       COMPUTES THE AMOUNT OF SEDIMENT DEPOSITED BY
*       HEMIPELAGIC RAIN AT AT INTERVALS OF DSTINC
*       METRES. THE SEDIMENT IS BROKEN DOWN BY GRAIN
*       SIZE.
*       THE RATE OF DEPOSITION IS CALCULATED AT A NUMBER
*       OF POINTS ACROSS THE FJORD, AND THE RESULTS ARE INTEGRATED
*       TO COMPUTE THE TOTAL FLUX.

      SUBROUTINE SEDFLX(PTS)

      REAL XB,DSTINC,Y(201),VX(201),VY(201),T(201),I0(4),PLUME
      REAL FLX(1001,4),LMDA(4),DAY2SC,CHANNL,CNTN(4),DEPTH
      INTEGER XBPNT,BINS,PTS,I,GRAINS,ROW
      COMMON /MATRIX/ FLX,GRAINS,DSTINC
       SAVE /MATRIX/
      COMMON /SED/ T,Y,VX,VY,LMDA,I0
      COMMON /FINDVX/ CHANNL,V0
*
*     READ THE PARAMETERS FROM USER
*     (FIRST READ PAST THE LINES WE DON'T NEED FOR THIS CALCULATION)
      REWIND 11
      DO 10,I=1,18
        READ(11,*,END=9900,ERR=9901)
10    CONTINUE

      DO 20,I=1,GRAINS
         READ(11,*,END=9900,ERR=9901)CNTN(I),LMDA(I)
```

```
20    CONTINUE

      READ(11,*,END=9900,ERR=9901)V0
      READ(11,*,END=9900,ERR=9901)CHANNL
      READ(11,*,END=9900,ERR=9901)DEPTH
      READ(11,*,END=9900,ERR=9901)PLUME
*     COMPUTE THE POSITION IN THE VECTOR OF THE POINT THAT REPRESENTS XB
*     THE PLUME IS ASSUMED TO SPREAD AT A 12 DEGREE ANGLE
      DG2RAD = 3.14159265358 / 180.0
      XB = 0.5 * (PLUME*1000.0 - CHANNL)/ TAN(12.0 * DG2RAD)
      XBPNT = INT(XB/DSTINC) + 1

      DAY2SC = 3600.0*24.0
      BINS = PTS - 1
*     CONVERT LMDA'S TO 1/SEC. COMPUTE INVENTORY
*     AND THE INITIAL FLUX IN EACH GRAIN SIZE.
*     SINCE, CNTN IS IN UNITS OF G/M**3, DIVIDE BY 1000.0 TO
*     CONVERT TO KG/M**3
      DO 50 I=1,GRAINS
         LMDA(I) = LMDA(I) / DAY2SC
         I0(I) = CNTN(I) * DEPTH / 1000.0
         FLX(1,I) = LMDA(I) * I0(I) * CHANNL
50    CONTINUE

      VX(1) = V0
        Y(1) = 0.0
      VY(1) =  0.0
      T(1) = 0.0
      DO 200 I=2,201
        Y(I) = Y(I-1) + CHANNL/400.0
        VX(I) = V0
        VY(I) = 0.0
        T(I) = 0.0
200   CONTINUE

*      COMPUTE FLX IN THE NEAR ZONE
      DO 500 ROW=2,XBPNT
        CALL FLXIN(ROW)
500   CONTINUE

      DO 600 ROW= XBPNT+1,PTS
        CALL FLXOUT(ROW)
600   CONTINUE

      GOTO 99999
9900  WRITE(*,'(''SUBROUTINE SEDFLX -- NO RECORD IN USER'')')
      GOTO 99999
9901  WRITE(*,'(''SUBROUTINE SEDFLX -- ERROR READING USER'')')
99999 CONTINUE
      RETURN
      END
***************************************************************
*     SUBROUTINE FLXIN
*     COMPUTE THE MASS OF EACH GRAIN SIZE DEPOSITED IN A CROSS SECTION
*     OF THE SEAFLOOR UNDER THE SPREADING PLUME
*     PNTFLX(1),T(1),Y(1),VX(1),VY(1) ARE ASSOCIATED WITH THE COLLECTION
*     POINT AT THE CENTRE OF THE PLUME. THE VECTORS REPRESENT COLLECTION
*     POINTS IN HALF A LATERAL CROSS-SECTION OF A SYMMETRICAL PLUME.
      SUBROUTINE FLXIN(ROW)
      REAL X,DSTINC,I0(4),LMDA(4),T(201),Y(201),VX(201),FLX(1001,4)
      REAL VY(201),A,B,PNTFLX(201),XBACK,TINC,VYINC,XFRD,HALFUP
      REAL VXBACK,VXFRD,E
      INTEGER PTSACC,BINACC,GRAINS,ROW,I,J
      COMMON /MATRIX/ FLX,GRAINS,DSTINC
      SAVE /MATRIX/
      COMMON /SED/ T,Y,VX,VY,LMDA,I0
      SAVE /SED/

      DATA E,PTSACC,BINACC/2.718281828,201,200/

      X = (ROW-1)*DSTINC
```

```
      XBACK = X - DSTINC
      XFRD = X + DSTINC

*     FOR EACH COLLECTION POINT IN THE HALF CROSS-SECTION, COMPUTE THE
*     SEDIMENT DEPOSITED.
      DO 100 I=1,PTSACC
*     TINC = TIME TO CROSS ONE GRID SQUARE TOWARD SEA
        TINC = DSTINC/VX(I)
*       COMPUTE NEW LATERAL POSITION AND ACCUMULATED TIME.
        Y(I) = Y(I) + TINC*VY(I)
        T(I) = T(I) + TINC

*       THERE IS NO LATERAL VELOCITY COMPONENT AT THE CENTRE OF THE
*       PLUME. IF I>1, CALCULATE THE LATERAL VELOCITY COMPONENT USING THE
*       APPROXIMATION:
*         (CHANGE IN VY) / TSTEP = - (CHANGE IN VX) / TSTEP
*         VX IS COMPUTED BY FUNCTION FNVX
        IF (I .GT. 1) THEN
          HALFUP = (Y(I)+Y(I-1))/2.0
          VXBACK = FNVX(XBACK,HALFUP)
          VXFRD = FNVX(XFRD,HALFUP)
          VYINC = -(VXFRD - VXBACK)*(Y(I)-Y(I-1))/(2.0*DSTINC)
          VY(I) = VY(I-1) + VYINC
        ENDIF
        VX(I) = FNVX(X,Y(I))
100     CONTINUE

*       COMPUTE THE TOTAL FLX DEPOSITED IN THE CROSS-SECTION
        DO 200 I=1,GRAINS
*         COMPUTE THE FLX OF ONE GRAIN SIZE DEPOSITED AT EACH POINT IN
*         THE HALF CROSS-SECTION
          A = LMDA(I) * I0(I)
          DO 150 J=1,PTSACC
            B = - LMDA(I) * T(J)
*           AVOID UNDERFLOW ERROR THAT OCCURS IF POWER OF E VERY SMALL
            IF (B .GT. -675.0) THEN
              PNTFLX(J) = A * (E**B)
            ELSE
              PNTFLX(J) = 0.0
            ENDIF
150       CONTINUE

*         INTEGRATE FLX IN THE HALF CROSS-SECTION AND DOUBLE
*         NOTE NO DIVISION BY 2 IS EQUIVALENT TO MULTIPLYING THE FLX
*         IN HALF THE PLUME BY 2.
          FLX(ROW,I) = 0.0
          DO 170 J=1,BINACC
            FLX(ROW,I) = FLX(ROW,I)
     C                   + (PNTFLX(J)+PNTFLX(J+1)) * (Y(J+1)-Y(J))
170       CONTINUE
200     CONTINUE
        RETURN
        END
**************************************************************************
*      FUNCTION FNVX
*
*      COMPUTES THE X-COMPONENT OF VELOCITY AT A POINT (X,Y)
*      IN A TWO DIMENTIONAL PLUME. THE ORIGIN IS AT THE CENTRE OF THE
*      DELTA HEAD RIVER CHANNEL. X IS LONGITUDINAL POSTION,Y IS THE
*      THE LATERAL POSITION,
       REAL FUNCTION FNVX(X,Y)
       REAL CHANNL,X0,X,BORDER,Y,K,C,PI,E,V0
       COMMON /FINDVX/CHANNL,V0
       SAVE /FINDVX/
       DATA C,PI,E/.109,3.14159265,2.718281828/
       X0 = CHANNL / (SQRT(PI)*C)
       IF(X.LT.X0)THEN
         BORDER = (1.0-X/X0)*CHANNL/2
         IF(Y.GT.BORDER)THEN
           K = -(Y + (X*C*SQRT(PI) - CHANNL)/2.0 )**2.0
           K = K/(2*C*C*X*X)
```

```
           FNVX = V0* (E**K)
         ELSE
           FNVX = V0
         ENDIF
       ELSE
         K = -(Y*Y/(2.0*C*C*X*X))
         FNVX = V0 * SQRT(X0/X) * (E**K)
       ENDIF
       RETURN
       END
************************************************************************
*     SUBROUTINE FLXOUT
*     COMPUTE THE MASS OF EACH GRAIN SIZE DEPOSITED IN A HALF CROSS SECTION
*     OF THE SEAFLOOR UNDER A PLUME OF CONSTANT WIDTH
*     PNTFLX(1),T(1),Y(1),VX(1),VY(1) ARE ASSOCIATED WITH THE COLLECTION
       POINT AT THE CENTRE OF THE PLUME. THE VECTORS REPRESENT COLLECTION
*    POINTS IN HALF A LATERAL CROSS-SECTION OF A SYMMETRICAL PLUME.
       SUBROUTINE FLXOUT(ROW)
       REAL FLX(1001,4)
       REAL DSTINC,I0(4),LMDA(4),T(201),Y(201)
       REAL A,B,PNTFLX(201),E,VX(201),VY(201)
       INTEGER BINACC,PTSACC,I,J,GRAINS,ROW
       COMMON /MATRIX/ FLX,GRAINS,DSTINC
       SAVE /MATRIX/
       COMMON /SED/ T,Y,VX,VY,LMDA,I0
       SAVE /SED/
       DATA E,PTSACC,BINACC/2.718281828,201,200/
       DO 100 I=1,PTSACC
          T(I) = T(I) + DSTINC/VX(I)
100    CONTINUE

*      COMPUTE THE TOTAL FLX IN THIS CROSS-SECTION FOR EACH GRAIN SIZE
       DO 200 I=1,GRAINS
*         COMPUTE THE FLX OF ONE GRAIN SIZE DEPOSITED AT EACH POINT IN
*         THE CROSS-SECTION
          A = LMDA(I) * I0(I)
          DO 150 J=1,PTSACC
            B = - LMDA(I) * T(J)
            IF (B .GT. -675.0) THEN
              PNTFLX(J) = A * (E**B)
            ELSE
              PNTFLX(J) = 0.0
            ENDIF
150       CONTINUE

*         INTEGRATE FLX IN THE HALF CROSS-SECTION AND DOUBLE
*         NOTE NO DIVISION BY 2 IS EQUIVALENT TO MULTIPLYING THE FLX
*         IN HALF THE PLUME BY 2.
          FLX(ROW,I) = 0.0
          DO 170 J=1,BINACC
            FLX(ROW,I) = FLX(ROW,I)
     C                        + (PNTFLX(J)+PNTFLX(J+1)) * (Y(J+1)-Y(J))
170      CONTINUE
200     CONTINUE
       RETURN
       END
```

USER DOCUMENTATION

SECTION 1 — BACKGROUND :

DELTA IS A LINEAR MODEL WHICH SIMULATES BASIN INFILLING THROUGH THE GROWTH AND PROGRADATION OF A DELTA. THE ASSUMPTIONS MADE ARE:

A) GROWTH IS DOMINATED BY A HEAD-WATER RIVER,
B) SEDIMENT ENTERING THE SYSTEM IS CONSTRAINED BETWEEN VERTICAL, PARALLEL INLET WALLS,
C) LATERAL VARIATIONS IN THE SHAPE OF THE DELTA ARE EPHEMERAL THE DELTA FRONT IS KEPT PERPENDICULAR TO THE BASIN WALLS.
D) THE LONGITUDINAL PROFILE OF THE BASIN HAS AN EXPONENTIAL SHAPE INITIALLY

SUSPENDED SEDIMENT:
DEPOSITION OF HEMIPELAGIC RAIN IS MODELLED BY DYNAMIC SIMULATION OF SEDIMENT TRANSPORT USING A VELOCITY FIELD FOR A TWO DIMENSIONAL RIVER JET ENTERING A STRATIFIED BASIN.

BEDLOAD:
CHANNELIZED SEDIMENT FLOWS ARE MODELLED BY SEPARATING PART OF THE BEDLOAD AS IT ENTERS THE BASIN, BYPASSING THE NEARSHORE AND DEPOSITING THE MATERIAL AS A FAN IN THE DEEP PART OF THE BASIN. THE REMAINING BEDLOAD IS DUMPED NEAR THE MOUTH OF THE RIVER AS DELTA FORESETS.

DIFFUSION:
PROCESSES, OTHER THAN TURBIDITY CURRENTS, WHICH MOVE SEDIMENT DOWNSLOPE, ARE MODELLED BY DIFFUSING SEDIMENT DOWNSLOPE USING A DIFFUSION EQUATION, BORROWED FROM HEAT DIFFUSION MODELS.

SECTION 2 — INPUT:

THE FILE "USER" CONTAINS THE INPUT PARAMETERS FOR THE MODEL. THE PURPOSE HERE IS TO DOCUMENT HOW THAT FILE IS USED.

THE LINE-BY-LINE DESCRIPTION IN THAT FILE SHOULD BE LEFT AS IS FOR REFERENCE. CHANGE THE VALUE AT THE LEFT OF EACH LINE TO SUIT YOUR SYSTEM.

*** COMMENT 1
THIS LINE MUST BE IN PLACE IN THE FILE, SINCE THE PROGRAM READS PAST IT. HOWEVER, IT HAS NO EFFECT ON THE RUN AND IS INTENDED AS A SPACE FOR MAKING NOTES ON THE PARTICULAR PARAMETERS.

590 INITIAL HEIGHT AT MODEL ORIGIN (M) 2
THE HEIGHT OF THE ORIGIN (HIGHEST POINT) ABOVE THE BOTTOM OF THE BASIN. USUALLY MAXIMUM DEPTH OF WATER IN THE BASIN. IF SEA LEVEL VARIES DURING THE RUN, THIS VALUE SHOULD BE AT LEAST AS GREAT AS THE MAXIMUM DIFFERENCE BETWEEN WATER LEVEL AND THE ORIGINAL DEPTH OF THE DEEPEST POINT IN THE BASIN.

100 DISTANCE FROM ORIGIN TO SILL (KM) 3
REAL LENGTH OF THE MODEL SITE

3.2 WIDTH OF FJORD BOTTOM IN KM 4

THE MODEL IS LINEAR, ASSUMING A RECTANGULAR CROSS-SECTION PERPENDICULAR TO THE BASIN WALLS. THIS IS THE WIDTH OF THAT RECTANGLE.

10 DISTANCE FROM HEAD TO TRACK SLOPE (KM) 5

PERIODICALLY SLOPE IS RECORDED A GIVEN DISTANCE FROM WHEREVER THE RIVER MOUTH IS AT THE TIME OF RECORDING. ENTER THAT DISTANCE HERE.

9000 TIME TO OUTPUT ACCUMULATION RATES ALONG PROFILE (YEARS) 6

WHEN THE MODEL HAS RUN FOR THIS MANY YEARS, IT WILL RECORD THE ACCUMULATION RATES AT INTERVALS ALONG THE PROFILE.

10 DISTANCE FROM ORIGIN WHERE RATES ARE TRACKED (KM) 7

PERIODICALLY, SLOPE AND ACCUMULATION RATES ARE RECORDED AT A GIVEN DISTANCE FROM THE ORIGIN. ENTER THAT DISTANCE HERE.

200 DISTANCE FROM FJORD HEAD OVER WHICH TO DUMP BEDLOAD (M) 8

THE BEDLOAD IS NOT DUMPED AS A PERFECT CUBOID SINCE THE DISCRETE NATURE OF THE MODEL WILL CAUSE A THICK, SHORT BLOCK TO ASSUME A RHOMBOIDAL PROFILE. THE BEDLOAD IS, HOWEVER DUMPED IN A SHAPE AS CLOSE TO A RECTANGLE AS POSIBLE. THIS PARAMETER IS BEST THOUGHT OF AS THE DISTANCE THIS RECTANGLE EXTENDS OUTWARD FROM THE RIVER MOUTH

T T=CONTROL BY TIME, D=CONTROL BY DISTANCE 9

THE LENGTH OF THE MODEL RUN AND THE INTERVAL BETWEEN PLOTS OF THE PROFILES CAN BE CONTROLLED EITHER BY TIME PASSED OR BY DISTANCE PROGRADED. FOR EXAMPLE, YOU CAN REQUEST THAT THE MODEL BE RUN FOR 30000 YEARS AND A PROFILE BE PLOTTED EVERY 5000 YEARS. IN THIS CASE ENTER A 'T' ON LINE 9. YOU MAY WANT TO PROGRADE THE MODEL 30 KM AND PLOT A PROFILE EVERY 5 KM. IN THIS CASE, ENTER A D ON LINE 9.

1 SLOPE AT WHICH BYPASS STARTS (DEGREES) 10

THE MODEL ASSUMES THERE IS A SLOPE CRITICAL TO THE DEPOSITION OF SEDIMENT GRAVITY FLOWS. IF THE FORESLOPE IS NOT AS STEEP AS THIS CRITICAL ANGLE, NO BYPASSING OCCURS.
BYPASS MATERIAL BEGINS TO BE DEPOSITED WHEN THE SLOPE OF THE PRODELTA IS LESS THAN THIS ANGLE.

590 INITIAL BASIN DEPTH IN METRES 11

INITIAL MAXIMUM DEPTH OF THE BASIN

********COMMENT 12

THE FIRST 9 PARAMETERS ARE FIXED FOR THE RUN. THE PARAMETERS FOLLOWING THIS COMMENT LINE MAY VARY. THE PROGRAM CHECKS FOR A NEW SET OF PARAMETERS AFTER IT IS FINISHED ITERATING THE REQUIRED TIME WITH THE CURRENT SET. IT WILL CONTINUE TO RUN UNTIL IT COLLECTS A DIFFUSION COEFFICIENT < 0.
THE INFORMATION ON THIS COMMENT LINE DOES NOT AFFECT THE MODEL, BUT THE LINE MUST BE PRESENT IN THE FILE.

15000 DIFFUSION COEFFICIENT (SQ.M/YR) 13

THE DIFFUSION COEFFICIENT DETERMINES THE RATE AT WHICH DEPOSITED SEDIMENT IS SMEARED DOWNSLOPE. THE LARGER THE DIFFUSION COEFFICIENT, THE FASTER MATERIAL IS MOVED. A DIFFUSION COEFFICIENT OF 0 STOPS ALL DIFFUSION FROM OCCURRING.

10000 YEARS TO RUN MODEL OR KMS TO PROGRADE WITH THESE PARMS 14

IF A "T" WAS ENTERED ON LINE 9, ENTER THE NUMBER OF YEARS TO RUN THE MODEL WITH THIS SET OF PARAMETERS. IF A "D" WAS ENTERED ON LINE 9, ENTER THE NUMBER OF KILOMETRES TO PROGRADE THE MODEL USING THIS SET OF PARAMETERS.

1000 YEARS OR KILOMETRES BETWEEN PLOTS 15

IF A "T" WAS ENTERED ON LINE 9, ENTER THE THE NUMBER OF YEARS BETWEEN PLOTS OF THE DELTA PROFILE. IF A "D" UAS ENTERED ON LINE 9, ENTER THE NUMBER OF KILOMETERS TO PROGRADE BETWEEN

```
          PLOTS

160             BEDLOAD (KG/SEC)                                              16
          RATE AT WHICH THE BEDLOAD ENTERS THE SYSTEM. HERE SYSTEM MEANS
          THE FORESLOPE OR BEYOND, DO NOT INCLUDE TOPSET DEPOSITS.

.3              FRACTION OF BEDLOAD TO BYPASS                                 17
         CHANNELIZED SEDIMENT FLOW IS MODELED BY TRANSPORTING A PORTION
         OF THE BEDLOAD TO THE PRODELTA. ENTER THE FRACTION OF THE BEDLOAD
         TO BYPASS.

50              DISTANCE OVER WHICH TO SPREAD BYPASS (KM)                     18
         THE MODEL SPREADS BYPASS MATERIAL FOR SOME DISTANCE OUT FROM
         THE POINT WHERE THE PRODELTA SLOPE FALLS BELOW THE CRITICAL
         ANGLE ENTERED ON LINE 9. ENTER THIS LONGITUDINAL DISTANCE
         ON THIS LINE. IF YOU WANT THE MATERIAL SPREAD BETWEEN THE
         CRITICAL POINT AND THE SILL, ENTER A DISTANCE GREATER THAN
         THE LENGTH OF THE MODEL. THE MODEL ASSUMES THAT GRAVITY FLOW
         MATERIAL CANNOT MOVE PAST THE SILL, SO THAT IF AN ATTEMPT IS MADE
         TO MOVE MATERIAL BEYOND THE SILL, THE DISTANCE IS
         READJUSTED SO THAT THE BYPASS MATERIAL IS DEPOSITED ENTIRELY
         IN THE BASIN


75.6,14.8,1750            : THESE FOUR LINES ARE FORMATTED INITIAL            19
28.4,5.6,1600             : CONCENTRATION(G/M**3)                             20
22.4,3.2,1500             : REMOVAL RATE CONSTANT(1/DAY), DENSITY(KG/M**3)    21
21.3,2.4,1400             :ONE LINE DEFINES ONE GRAIN SIZE                    22
         THE ABOVE FOUR LINES DESCRIBE THE SUSPENDED SEDIMENT LOAD AT
         THE RIVER MOUTH. VALUES ARE YEARLY AVERAGES.
         WE HAVE USED FOUR CATEGORIES AND THE HEADINGS IN THE
         OUTPUT REFLECT THESE CATEGORIES. THE FIRST LINE DESCRIBES
         COARSE SILT; THE NEXT, MEDIUM SILT; THE THIRD, FINE SILT; THE
         LAST, CLAY. THE MODEL MAY BE RUN WITH FEWER GRAIN SIZES BY
         ENTERING A CONCENTRATION OF 0.0 IN THE EXTRA CATEGORIES

1.3             VELOCITY (M/SEC) OF WATER AT RIVER MOUTH                      23
         AVERAGE VELOCITY OF RIVER WATER ENTERING FJORD.

100             WIDTH OF RIVER CHANNEL (M)                                    24
         THE MODEL ASSUMES THE CHANNEL OPENING IS A RECTANGULAR SLOT.
         THIS IS THE AVERAGE WIDTH OF THAT OPENING.

2.5            DEPTH OF RIVER CHANNEL (M)                                     25
         AVERAGE DEPTH OF RECTANGULAR OPENING.

2.438        MAXIMUM WIDTH OF RIVER PLUME (KM)                                26
         THE MODEL ASSUMES THE RIVER PLUME IS CONSTRAINED BY THE WALLS OF
         THE FJORD. THIS IS THE MAXIMUM WIDTH OF THE RIVER PLUME AFTER
         IT STOPS SPREADING LATERALLY.

*******COMMENT
      LINES 12 TO 26 MAY BE REPEATED INDEFINITELY TO CHANGE PARAMETERS
      DURING THE RUN. TO STOP THE MODEL, ENTER A NEGATIVE VALUE FOR
      DIFFUSION FOLLOWING THE COMMENT LINE

-1         DIFFUSION COEFFICIENT (SQ.M/YR)
```

SECTION 3 — OUTPUT:

PROGRADATION RATE AS A FUNCTION OF TIME

PROGRADATION RATES ARE PRINTED AT APPROXIMATELY 500 YEAR INTERVALS. TO CHANGE THE INTERVAL, CHANGE THE VALUE OF TOFT IN THE DATA STATEMENT IN SUBROUTINE CLENUP.

```
TIME        PROGRADATION RATE
```

```
 (YEARS)             (M/YR)
  512.50              2.70
 1016.50              2.44
 1518.00              2.38
 1998.00              2.27
 2492.00              2.17
 2997.00              2.17
 3510.50              2.13
 3983.50              2.13
 4509.00              2.08
 4991.00              2.08
 5476.00              2.08
 6013.00              2.08
 6503.00              2.00
 6994.50              2.04
 7488.50              2.04

ACCUMULATION RATES AND SLOPE 10.00 KM FROM ORIGIN

   THIS IS THE OUTPUT FOR THE DISTANCE REQUESTED ON LINE 7 OF THE
   INPUT FILE. TO CHANGE THE TIME INTERVAL BETWEEN RECORDINGS,
   CHANGE THE VALUE OF TRKOFT IN THE DATA STATEMENT OF SUBROUTINE
   ITRATE.

 TIME    SLOPE     HEMIPELAGIC HEMIPELAGIC  TOTAL
 (YEARS) (DEGREES) RAIN        RAIN AND     ACCUMULATION
                   (CM/A)      BYPASS       (CM/A)
                               (C/A)
     3.7   1.1      .8         1.4         2.7
   203.5   1.1      .9         1.5         3.3
   403.6   1.1     1.1         1.7         3.6
   603.4   1.2     1.2         1.8         4.0
   803.5   1.3     1.3         2.0         4.4
  1003.7   1.3     1.5         2.1         4.8
  1203.5   1.4     1.7         2.4         4.9
  1403.6   1.4     1.9         2.6         5.4
  1603.4   1.5     2.2         2.8         5.7
  1803.5   1.6     2.4         3.1         5.8
  2003.7   1.6     2.6         3.3         6.2
  2203.5   1.7     2.8         3.5         6.4
  2403.6   1.7     3.0         3.6         6.6
  2603.4   1.8     3.1         3.8         6.8
  2803.5   1.9     3.3         4.0         7.0
  3003.7   1.9     3.4         4.0         7.2
  3203.5   2.0     3.4         4.0         7.3
  3403.6   2.0     3.3         3.9         7.6
  3603.4   2.1     3.0         3.7         7.7
  3803.5   2.1     2.7         3.4         7.1
  4003.7   1.9     2.4         3.1         7.1

SLOPE 10.00 KM FROM RIVER MOUTH AT TIME INTERVALS

  THE SLOPE AT THE DISTANCE FROM THE RIVER MOUTH REQUESTED
  IN LINE 5 OF THE INPUT FILE. THIS IS RECORDED AT THE SAME
  INTERVALS AS THE ACCUMULATION RATES ABOVE AND THE INTERVAL
  IS CHANGED IN THE SAME WAY.

 TIME    SLOPE
 (YEARS) (DEGREES)
     3.7   1.1
   203.5   1.0
   403.6   1.0
   603.4   1.0
   803.5   1.0
  1003.7   1.0
  1203.5   1.0
  1403.6   1.1
  1603.4   1.1
  1803.5   1.1
```

```
    2003.7   1.1
    2203.5   1.1
    2403.6   1.1
    2603.4   1.1
    2803.5   1.1
    3003.7   1.1
    3203.5   1.1
    3403.6   1.1
    3603.4   1.1
    3803.5   1.1
    4003.7   1.1
    4203.5   1.1
    4403.6   1.1
    4603.4   1.1
    4803.5   1.1
    5003.7   1.1

 ACCUMULATION RATES AFTER  9003.7 YEARS

    ACCUMULATION RATES ARE PRINTED FOR ABOUT 30 POINTS ALONG THE
    PROFILE. THE NUMBER OF POINTS MAY BE CHANGED BY CHANGING THE
    VALUE OF NUMPTS IN THE DATA STATEMENT OF SUBROUTINE ITRATE

 DISTANCE HEMIPELAGIC HEMIPELAGIC TOTAL
 (KM)       RAIN        RAIN AND    ACCUMULATION
           (CM/A)       BYPASS      (CM/A)
                        (CM/A)
    2.7      3.3          4.0         6.5
    5.3      2.3          2.9         5.5
    7.9      1.2          1.8         4.5
   10.5       .7          1.3         3.6
   13.1       .5          1.1         3.0
   15.7       .3           .9         2.4
   18.3       .3           .8         2.0
   20.9       .2           .7         1.6
   23.5       .2           .7         1.4
   26.1       .1           .6         1.2
   28.7       .1           .6         1.0
   31.3       .1           .5          .9
   33.9       .1           .5          .8
   36.5       .0           .5          .7
   39.1       .0           .5          .6
   41.7       .0           .4          .6
   44.3       .0           .4          .5
   46.9       .0           .4          .5
   49.5       .0           .4          .5

 PARTICLE SIZE FRACTIONATION WITH DISTANCE FROM THE RIVER MOUTH

    THIS LISTING GIVES THE PROPORTION BY MASS OF EACH GRAIN SIZE
    DEPOSITED BY HEMIPELAGIC RAIN AT VARIOUS DISTANCES FROM THE RIVER
    MOUTH.
    TO CHANGE THE INTERVALS AT WHICH THE DATA IS PRINTED, CHANGE THE
    INCREMENTATION OF I IN THE IF STATEMENT AT THE END OF THE LOOP IN
    SUBROUTINE DSTNCE OF PROGRAM GRAIN. (SOURCE FILE IS GRNTXT).

 DISTANCE CS   MS   FS   CLAY
    (KM)  (%)  (%)  (%)  (%)
      .0  80   11    5    4
     1.0  78   12    6    4
     2.0  75   14    7    5
     3.0  71   15    8    6
     4.0  66   18    9    7
     5.0  59   20   12    9
     6.0  52   23   14   11
     7.0  44   26   17   14
     8.0  37   28   19   16
     9.0  30   30   22   19
    10.0  23   31   25   21
```

```
  15.0   6   29   33   32
  20.0   1   23   36   40
  25.0   0   16   37   46
  30.0   0   11   37   51
  35.0   0    8   36   56
  40.0   0    5   35   60
  45.0   0    4   33   64
  50.0   0    2   31   67
  55.0   0    2   28   70
  60.0   0    1   26   73
  65.0   0    1   24   75
  70.0   0    0   22   77
  75.0   0    0   20   79
  80.0   0    0   18   81
  85.0   0    0   17   83
  90.0   0    0   15   85
  95.0   0    0   14   86
 100.0   0    0   12   88
```

PARTICLE SIZE FRACTIONATION AT 10. KM

THIS LISTING GIVES THE PROPORTION OF EACH GRAIN SIZE DEPOSITED AS A FUNCTION OF TIME. THE DISTANCE, 10.0 KM, IS READ FROM LINE 7 OF THE INPUT FILE AND IS MEASURED FROM THE ORIGINAL RIVER MOUTH OF THE MODEL.
TO CHANGE THE TIME INTERVAL OF THE OUTPUT, CHANGE TINC IN THE DATA STATEMENT OF SUBROUTINE TIMINT OF PROGRAM GRAIN. (SOURCEFILE,GRNTXT).
******THIS PARTICULAR OUTPUT IS VALID ONLY IF THE MODEL IS RUN WITH CONSTANT VALUES IN LINES 19 THROUGH 26 OF THE INPUT FILE. SUBROUTINE TIMINT DOES NOT HANDLE DYNAMIC SPM CONDITIONS.*******

```
 TIME    CS    MS    FS    CLAY
        (%)   (%)   (%)   (%)
    0.   23    31    25    21
  200.   29    30    22    19
  400.   32    29    21    18
  600.   36    28    20    16
  800.   39    27    18    15
 1000.   42    26    17    14
 1200.   46    25    16    13
 1400.   49    24    15    12
 1600.   52    23    14    11
 1800.   56    21    13    10
 2000.   59    20    12    09
 2200.   62    19    11    08
 2400.   65    18    10    07
 2600.   67    17    09    07
 2800.   69    16    08    06
 3000.   71    15    08    06
 3200.   73    15    07    05
 3400.   74    14    07    05
 3600.   76    13    06    05
 3800.   77    13    06    04
 4000.   78    12    06    04
 4200.   79    12    05    04
 4400.   79    12    05    04
```

PROFILES

THE PROGRAM GENERATES A FILE(PLFILE) OF PROFILES USING THE PLOTTING PACKAGE DISSPLA. THE FILE IS PLOTTED USING THE ATTACH, POPZETA, AND PLOTOFF COMMANDS IN THE BATCH JOB FILE.

SECTION 4 — FILES

USERDOC : USER DOCUMENTATION
DELTXT : SOURCE FILE FOR THE DELTA MODEL
GRNTXT : SOURCE FILE FOR THE PARTICLE SIZE ANALYSIS MODEL
DELJOB — JOB FILE (NOS)
DELFILE — OUTPUT FILE FROM MODEL RUN

* * *

SECTION 5 — CHANGING SEA LEVEL

TO VARY SEA LEVEL, REPLACE THE STATEMENT FUNCTION LEVEL(T) = .0 IN THE MAINLINE OF PROGRAM DELTA(SOURCE FILE, DELTXT). LEVEL(T) MUST EXPRESS SEA LEVEL AS A FUNCTION OF TIME, T, WHERE LEVEL(0)=0. LEAVING THE FUNCTION AS IT IS RESULTS IN THE SEA LEVEL BEING FIXED FOR THE DURATION OF THE RUN.

SECTION 6 — SUBMITTAL

IF YOU'RE USING THE BIO CYBER, EDIT THE FILE, USER, CHANGING ANY PARAMETERS THAT DO NOT REFLECT CONDITIONS IN THE SYSTEM YOU'RE MODELLING. CHANGE THE NAME AND EXT PARAMETERS IN THE PLOTOFF COMMAND IN DELJOB. REPLACE THE FILES AND SUBMIT DELJOB. THE JOB PRODUCES A REPORT FILE, DELFILE, WHICH IT ROUTES TO THE PRINTER. IT ALSO PRODUCES A PLOT ON THE ZETA. IF YOU'RE RUNNING THE JOB ON SOME OTHER INSTALLATION, THE JOB FILE SHOULD BE CHECKED FOR PORTABILITY TO YOUR SYSTEM.

Example of input file :

```
*** COMMENT                                                                       1
438.0           INITIAL HEIGHT AT MODEL ORIGIN (M)                                2
100             DISTANCE FROM ORIGIN TO SILL (KM)                                 3
4.38            WIDTH OF FJORD BOTTOM IN KM                                       4
20              DST FROM HEAD TO TRACK SLOPE (KM)                                 5
1500            TIME TO OUTPUT ACCUMULATION RATES ALONG PROFILE (YEARS)           6
10              DISTANCE FROM RIVER MOUTH WHERE RATES ARE TRACKED (KM)            7
500
T               T=CONTROLTBY TIME, D=CONTRL BY DISTANCE                           9
1               SLOPE AT WHICH BYPASS STARTS (DEGREES)                           10
438.0           SEA LEVEL  (HEIGHT OF WATER IN M)                                11
********COMMENT                                                                  12
14000           DIFFUSION COEFFICIENT (SQ.M/YR)                                  13
10000           YEARS TO RUN MODEL OR KMS TO PROGRADE WITH THESE PARMS           14
2000            YEARS OR KILOMETRES BETWEEN PLOTS                                15
135             BOTTOM LOAD (KG/SEC)                                             16
.3              FRACTION OF BOTTOM LOAD TO BYPASS                                17
50              DISTANCE OVER WHICH TO SPREAD BYPASS (KM)                        18
76.9,19.4,1750              : THESE FOUR LINES ARE FORMATTED INITIAL             19
76.9,7.5,1600               : CONCENTRATION(G/M**3)                             20
76.9,4.2,1500               : REMOVAL RATE CONSTANT(1/DAY), DENSITY(KG/M**3)     21
76.9,3.1,1400               :ONE LINE DEFINES ONE GRAIN SIZE                     22
1.3             VELOCITY (M/SEC)                                                 23
158.1           WIDTH OF RIVER CHANNEL (M)                                       24
1.58           DEPTH OF RIVER CHANNEL (M)                                        25
3.285        MAXIMUM WIDTH OF RIVER PLUME (KM)                                   26
```