

SANDIA REPORT

SAND2008-5621

Unlimited Release

Printed September 2008

Sandia National Laboratories Environmental Fluid Dynamics Code: Sediment Transport User Manual

Phi Hung X. Thanh, Matthew D. Grace, and Scott C. James

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.



Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online order: <http://www.doe.gov/bridge>

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: <http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online>



SAND2008-5621
Unlimited Release
Printed September 2008

Sandia National Laboratories Environmental Fluid Dynamics Code: Sediment Transport User Manual

Phi Hung X. Thanh
Sandia National Laboratories
Thermal/Fluids Science and Engineering Department
P.O. Box 969
Livermore, CA 94551-0969
phthanh@sandia.gov

Matthew D. Grace
Sandia National Laboratories
Thermal/Fluids Science and Engineering Department
P.O. Box 969
Livermore, CA 94551-0969
mgrace@sandia.gov

Scott C. James
Sandia National Laboratories
Thermal/Fluids Science and Engineering Department
P.O. Box 969
Livermore, CA 94551-0969
scjames@sandia.gov

Abstract

This document describes the sediment transport subroutines and input files for the Sandia National Laboratories Environmental Fluid Dynamics Code (SNL-EFDC). Detailed descriptions of the input files containing data from Sediment Erosion at Depth flume (SEDflume) measurements by Roberts et al. (2003) are provided, along with a description of the transport model developed by Jones (2001; Jones and Lick 2001), referred to in this document as SEDLJZ. This model is unique in that it directly incorporates SEDflume data. Both a theoretical description of sediment transport employed in SNL-EFDC and an explanation of the source code are provided. This user manual is meant to be used in conjunction with the original EFDC manual (Hamrick 1996). As such, a description about the hydrodynamics of EFDC is not provided in this report.

Through this document, the authors aim to provide the necessary information for users who wish to implement sediment transport in EFDC and obtain a clear understanding of the source code.

Acknowledgement

The research described in this document was funded through Sandia National Laboratories' Enabling Predictive Simulation Research Institute and Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Table of Contents

<i>Abstract</i>	4
<i>Acknowledgement</i>	4
<i>Introduction</i>	7
<i>SEDZLJ Sediment Input Files</i>	8
ERATE.SDF	8
BED.SDF	11
<i>SEDZLJ Sediment Source Codes</i>	13
S_SEDIC.f90	13
SCANSEDZLJ.f90	18
S_MAIN.f90	19
S_SHEAR.f90	23
S_BEDLOAD.f90	28
S_SEDZLJ.f90	32
S_MORPH.f90	44
<i>Variable Dictionary</i>	46
<i>Special Notes</i>	51
<i>References</i>	52

List of Figures

Figure 1 : The input file for erosion rate data from SEDflume: <code>erate.sdf</code>	9
Figure 2: The input file for sediment bed cores of each cell in the entire domain: <code>core_field.inp</code>	10
Figure 3: The input file for sediment bed layer information: <code>bed.sdf</code>	12

Introduction

This document describes the Sandia National Laboratories Environmental Fluid Dynamics Code (SNL-EFDC), which is a modified version of the EPA's public-domain surface-water flow, sediment transport, and water-quality model developed by John Hamrick while at the Virginia Institute of Marine Sciences (Hamrick 1992). EFDC simulates flow and transport of sediment (in bedload and in suspended load), algae (including growth kinetics), and toxic substance (kinetic reactions and transport). SNL-EFDC improves EFDC with updated sediment kinetics subroutines (e.g., sediment exchange with the bed) developed by Jones (2001; Jones and Lick 2001). Sediment erosion is calculated using data collected with a Sediment Erosion at Depth flume (SEDflume). SEDflume measures erosion rates as a function of shear stress and depth from relatively undisturbed cores taken directly from the sediment bed below the water body of interest. The use of SEDflume data provides more accurate sediment erosion rates that are directly input to the model.

The SNL-EFDC code includes the following subroutines: `s_main.f90`, `s_sedic.f90`, `s_shear.f90`, `s_bedload.f90`, `s_morph.f90`, `s_sedzlj.f90`, and `SCANSEDZLJ.f90`. The main subroutine (`s_main.f90`) calls the other subroutines, orchestrates the numerical sediment transport process, and outputs the sediment flux data to the main EFDC code. The initializer subroutines (`s_sedic.f90` and `SCANSEDZLJ.f90`) read data from the input files and convert the SEDflume data into EFDC variables. The shear subroutine (`s_shear.f90`) calculates the shear stress from the velocity profile output of the hydrodynamics portion of EFDC and any wind- and wave forcing functions. This shear stress information is used to determine the erosion and deposition processes in the sediment transport. The bedload subroutine (`s_bedload.f90`) calculates the probability of a sediment particle suspended in the water column or transported as saltating bedload. It also calculates the sediment bedload flux and the bedload sediment concentration. The primary sediment dynamics subroutine is the SEDZLJ subroutine (`s_sedzlj.f90`), which calculates the erosion and deposition of the sediment layers, determines the change in the bed layer's thickness, and incorporates the data read in by `s_sedic.f90` in sediment transport. Lastly, the morphology subroutine (`s_morph.f90`) allows for the incorporation of the changing hydrodynamic boundary conditions due to sediment erosion and deposition. These sediment transport subroutines use three input files that contains SEDflume data. These input files (`erate.sdf`, `bed.sdf`, and `core_field.sdf`) are read in by `s_sedic.f90` and `SCANSEDZLJ.f90`. Detailed description and examples for these input files are given below. In addition, a variable dictionary is provided at the end of the report.

SEDZLJ Sediment Input Files

ERATE.SDF

The example input file `erate.sdf` shown in Figure 1 contains erosion rate data from SEDflume. First, data about the thickness of each of the bed layer is provided in centimeters. The first two bed thicknesses are always zero (0.0) to account for the active and deposited layers (this point is described later, e.g., the section on `S_SEDZLJ.F90`). The top two layers will have thicknesses if there is deposition or armoring, otherwise they will have no thickness (when the dynamics are purely erosive). Thicknesses may increase or decrease according to subsequent deposition and erosion. Note that the minimum number of layers is 3. A dimensionless multiplier for the active layer thickness (TACTM) follows the layer thicknesses. In addition, `erate.sdf` can have multiple core samples (although data from only a single core appears in Figure 1). The sediment bed will have different sets of core data as specified in `core_field.sdf` with integer identification (1, 2, 3, etc.) corresponding to the order of each core's data set in `erate.sdf`. As shown in the `core_field.sdf` file in Figure 2, the numbers 1 and 2 correspond to the first and second data tables in `erate.sdf`, respectively (note that only the first line of core 2 is shown). Within each table, the critical shear stress values are listed for each of the bed layers (4.25 dynes/cm² in our example). The number of bed layers is specified in Card 36 of the main input file `efdc.inp`, as well as in `bed.sdf`. Critical shear stress for erosion (in dynes/cm²) is the minimum shear stress necessary for erosion to occur. The next set of inputs describes the bed or sediment core density, which accounts for both solid sediment density and porosity of sediment particles. The water density and the solid sediment density are listed next. In the example in Figure 1, the water density is reported at STP as 1.0 g/cm³ and the solid sediment density is for quartz at 2.6 g/cm³. The particle size distribution as a percentage is listed after the sediment density information. In the file `efdc.inp`, the number of particle sizes, NSED (also referred to as the number of particle size classes), is specified in both Card 22 (`efdc.inp`) and in `bed.sdf`. Next, erosion rate data are listed. The erosion rate data start with the value of the applied shear and the following line provides the erosion rate for each sediment layer at that shear stress. In Figure 1, the applied shear stresses are 0, 2, 4, 8, 10, and 20 Pa, and the erosion rates are listed in cm/s. For multiple core samples, critical shear stress, bulk density, particle size distribution, and initial bed erosion rates are reported in the same format, below the first set of core sample data.


```

*****
# Layer Thicknesses (cm) #
0.0 0.0 15.0 15.0 15.0
# Active Layer Thickness Multiplier #
5.0
# Critical Shear Stress for Erosion(dynes/cm^2) #Data for sediment 1
4.25 4.25 4.25 4.25 4.25
# Bed core Density (g/cm^3) #
1.9 1.9 1.9 1.9 1.9
# Water Density (g/cm^3) and Sediment Density (g/cm^3) #
1.0 2.6
# Particle Size Distribution (%) #
2.955 6.649 12.664 13.493 14.679 27.094 9.223 10.231 3.184
2.955 6.649 12.664 13.493 14.679 27.094 9.223 10.231 3.184
2.955 6.649 12.664 13.493 14.679 27.094 9.223 10.231 3.184
2.955 6.649 12.664 13.493 14.679 27.094 9.223 10.231 3.184
2.955 6.649 12.664 13.493 14.679 27.094 9.223 10.231 3.184
# Initial Bed Erosion Rates (cm/s) #
0.0 !shear stress 1
0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 !erosion rate of each layer
2.00000 !shear stress 2
0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00
4.00000 !shear stress 3
8.000E-05 8.000E-05 8.000E-05 8.000E-05 8.000E-05
8.00000 !shear stress 4
7.000E-03 7.000E-03 7.000E-03 7.000E-03 7.000E-03
10.0000 !shear stress 5
1.140E-02 1.140E-02 1.140E-02 1.140E-02 1.140E-02
20.0000 !shear stress 6
0.649E-01 0.649E-01 0.649E-01 0.649E-01 0.649E-01
# Critical Shear Stress for Erosion (dynes/cm^2) #Data for sediment 2 (same
structure as above)
*****

```

Figure 1 : The input file for erosion rate data from SEDflume: erate.sdf.

CORE FIELD.SDF

The `core_field.sdf` file specifies which core data set listed in `erate.sdf` corresponds to which cells of the bed. The first entry in the file is the number of different data sets (or the number of different data cores collected). After the number of cores, a matrix with the core identification number for each cell (defined in the `cell.inp` and `celllt.inp` files). Note that the core data exist for water cell (active cells) and non-water cells. In the example `core_field.inp` file in Figure 2, the model has 9 rows and 15 columns.

```
*****
2 !Number of cores
2 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 1 1 1 1 1 1 1 1 1 1 1 1
*****
```

Figure 2: The input file for sediment bed cores of each cell in the entire domain: `core_field.inp`.

BED.SDF

The input file `bed.sdf` contains information about the sediment bed layers. The first line contains names of the model parameter flags, whose values appear in the second line. When `VAR_BED = 1`, variable sediment bed option is activated, resulting in the usage of data from `core_field.sdf`, allows the model to incorporate spatially varying sediment characteristics. The `BEDLOAD` flag for bedload transport appears next. `NEQUIL` is the flag that activates adsorption of toxics onto sediments. `KB` is the maximum number of sediment layers, and should coincide with the `KB` value in Card 36 of the `EFDC.INP` file. `ISEDTIME` is the number of time steps before sediment transport is activated. This option allows the velocity profile to develop before sediment transport occurs. The `ISMORPH` flag activates morphology calculations for cases when erosion or deposition of sediment on the bed significantly changes the geometry of the system. `ISMORPH` incorporates the change in geometry into the hydrodynamic model. `IFWAVE` is the flag that turns on wind-driven waves that introduce additional shear at the sediment bed. `MAXDEPLIMIT` is the maximum fraction of sediment in the bottom water layer allowed to deposit onto the sediment bed during a single time step (it should usually be 1.0 unless instabilities are observed). `ITBM` and `NSICM` are array parameters read by `SCANSEDZLJ.F90`. `ITBM` is the number of different `SEDflume` shear-stress interpolants used to calculate erosion of newly deposited sediments. These values are listed `erate.sdf` and in the example from Figure 3, they have values 0, 2, 4, 8, 10, 20 Pa. `NSICM` is the number of size classes of interpolants used to estimate erosion rates of newly deposited sediment (described below). `ZBSKIN` (μm) is the roughness parameter. This parameter accounts for both grain-size roughness and bed-form roughness (topographical variations from dunes and ripples at the bottom of the water body). If `ZBSKIN` is zero, then each model cell's roughness is the local average particle (μm), the average particle diameter in the cell. `TAUCONST` (dynes/cm^2) is an option that, when greater than zero, allows the user to set the shear stress to a constant value in the flow. If `TAUCONST` is zero, the stresses are calculated from the velocity profile resulting from the hydrodynamic model. The next line of data contains the discretized size sediment classes in units of micrometers. In the example file in Figure 3, there are nine different size classes used to approximate the sediment distribution for the model of interest. For each of these size classes, the discretized diameter (`D50`; μm) is listed, and values of critical shear for erosion (`TCRE`; dynes/cm^2) and critical shear for suspension (`TCRSUS`; dynes/cm^2) in units of dynes/cm^2 are reported in the following lines. The critical shear for erosion is calculated according to Soulsby (1997) and the critical shear for suspension is calculated according to equations (8) and (9) in Van Rijn (1984b). These two values are used to calculate the fraction of sediment in suspended load versus bedload. The rest of the `bed.sdf` input file contains information for newly deposited sediments from `SEDflume`. Newly deposited sediments are defined as those that have undergone erosion (and perhaps suspension) and have then settled on the bed again. The next line of data provides the `NSICM` interpolant sediment size classes, `SCLOC(1:NSICM)` (μm). For each interpolant size class (`SCLOC`), the critical shear stress for erosion (`TAUCRITE`; dynes/cm^2) is provided. These values are either site-specific data from `SEDflume` or can be calculated according to equations (6) or (9) and (10) from Roberts et al. (1998). Lastly, a table of erosion rate data for the newly deposited sediment is listed (`ENRATE`; cm/s). `ENRATE` is calculated for each `SCLOC` at the shear values listed in `erate.sdf` (0, 2, 4, 8, 10, and 20 Pa). These are typically site-specific data or from Table 1 of Robert et al. (1998).

```

*****
# VAR_BED Bedload Nequil KB ISEDTIME IMORPH IFWAVE MAXDEPLIMIT#
  1      1      0      5      100      0      0      1.0
# ITBM NSICM Array Parameters #
  6      8
# ZBSKIN (>0 sets z0 in [um]) TAUCONST [dynes/cm^2] (> 0 to set constant) #
  1500      10
# Discretized Size Classes D50(K) [um] #
237.7 427.0 603.5 853.5 1070.5 1570.5 2415.0 3415.0 5450.1
# Critical Shear for Erosion TCRE(K) [dynes/cm2] #
1.5 2.4 3.3 4.25 7.6 9.5 10.8 16. 24.8 !Soulsby 1997
# Critical Shear for Suspension TCRSUS(K) [dynes/cm2] #
1.5 2.6 4.5 21.2 53.6 67.3 87.9 122.6 192. !van Rijn 1984b Eqs. (8) and (9)
# Sediment Bed Size (um) Tables #
125.0 222.0 432.0 1020.0 2000.0 2400.0 3000.0 6000.0
# Shear for Erosion of Bed Surface (dynes/cm^2) #
1.20 2.27 2.96 4.17 5.46 5.88 6.42 8.48 !Robert et al. 1998
                                           eqn (6) or (9) and (10)
# Erosion rates for active and deposited Layers (cm/s) # !Robert et al.
                                           1998 Table 1
1E-9 6.60E-5 4.66E-4 3.29E-3 6.17E-3 4.36E-2 !125 um !These six values
are for the 6 shear stresses specified in ITBM and defined in erate.sdf as
the initial bed erosion rates.
1E-9 5.97E-5 5.97E-4 5.96E-3 1.25E-2 1.25E-1 !222 um
1E-9 3.65E-4 2.16E-3 1.27E-2 2.25E-2 1.33E-1 !432 um
1E-9 2.01E-4 1.14E-3 6.51E-3 1.14E-2 6.49E-2 !1020 um
1E-9 2.25E-5 1.63E-4 1.19E-3 2.25E-3 1.63E-2 !2000 um
1E-9 1.12E-5 8.40E-5 6.28E-4 1.20E-3 8.98E-3 !2400 um
1E-9 3.96E-6 3.07E-5 2.28E-4 4.59E-4 3.56E-3 !3000 um
1E-9 2.03E-8 1.67E-7 1.44E-6 2.88E-6 2.49E-5 !6000 um
*****

```

Figure 3: The input file for sediment bed layer information: bed.sdf.

SEDZLJ Sediment Source Codes

S_SEDIC.f90

This routine reads in the three input files and calculates several sediment transport variables. First, s_sedic.f90 opens the three input files:

```
OPEN(UNIT=10,FILE='erate.sdf')
OPEN(UNIT=20,FILE='core_field.sdf')
OPEN(UNIT=30,FILE='bed.sdf')
```

Variables in bed.sdf are read first. The first two lines after the line containing STR_LINE contain flag options. After these flags, the roughness and shear stress are loaded. After the shear stress comes the discretized particle size classes (D50). The critical shear stresses for erosion (TCRE) and critical shear stresses for suspension (TCRSUS) for each sediment size class (D50) are read next. The file bed.sdf will also be read later for newly deposited sediment transport information. Note that NSCM is the number of sediment size classes (from EFDC), and should equal NSICM.

```
READ (30,'(A80)') STR_LINE
READ (30,*) VAR_BED,NCALC_BL,NEQUIL,KB,ISEDTIME,IMORPH_SEDZLJ,IFWAVE
IF(IMORPH_SEDZLJ==1.AND.(IMORPH/=0.OR.ISGWIT>=2)) THEN
    PRINT*, 'EFDC and SEDZLJ MORPHOLOGIES BOTH TURNED ON'
    STOP
ENDIF
READ (30,'(A80)') STR_LINE !SKIP THESE LINES
READ (30,'(A80)') STR_LINE !DATA READ BY SCANSEDZLJ; ITBM and NSICM.
READ (30,'(A80)') STR_LINE
READ (30,*) ZBSKIN,TAUCONST
READ (30,'(A80)') STR_LINE
READ (30,*) (D50(K),K=1,NSCM)
READ (30,'(A80)') STR_LINE
READ(30,*) (TCRE(K),K=1,NSCM)
READ (30,'(A80)') STR_LINE
READ (30,*) (TCRSUS(K),K=1,NSCM)
```

After obtaining the first portion of data in bed.sdf, s_sedic.f90 then reads core_field.sdf. The number of different core samples and the core sample corresponding to each cell in the domain is read.

```
READ(20,*) INCORE !read the number of cores
DO J=JC,1,-1
    READ(20,*)(NCORENO(I,J),I=1,IC)
ENDDO
```

Next, the routine loads core sample data from erate.sdf to determine the erosion rate before deposition or sorting occurs. TAUTEMP (dynes/cm²) contains the critical shear stresses necessary to erode each sediment layer. BDEN (g/cm³) is the bulk density for the bed layers; this is used in conjunction with bed thickness to define bed layer thickness (TSED, TACT, etc., all in units of g/cm²). WATERDENS and SEDDENS are the water density and solid sediment density

(g/cm³). PNEW is the mass percentage of each sediment size class in each layer, expressed as a percentage, but the form of PNEW in the code is the fraction PER (PER = PNEW/100.0). Lastly, TAULOC (dynes/cm²) and EORATE (cm/s) are listed, with TAULOC being the applied shear and EORATE the erosion rate of each sediment layer subject to the applied TAULOC. EORATE is the erosion rate of the site-specific (original bed) data versus ENRATE from `bed.sdf`, which is the erosion rate of newly deposited sediment.

```

DO CORE=1, INCORE !for each core of data
  READ (10, '(A80)') STR_LINE
  READ (10, *) (TAUTEMP(CORE, LL), LL=1, KB) !read the critical shear stresses of
                                         the core

  READ (10, '(A80)') STR_LINE
  READ (10, *) (BDEN(CORE, LL), LL=1, KB) !read in the bulk density of the core
  READ (10, '(A80)') STR_LINE
  READ (10, *) WATERDENS, SEDDENS !read in the water density and sediment
                                   solid's density

  READ (10, '(A80)') STR_LINE
  DO LL=1, KB
    READ (10, *) (PNEW(CORE, LL, K), K=1, NSCM)
  ENDDO
  READ (10, '(A80)') STR_LINE
  DO K=1, ITBM
    READ (10, *) TAULOC(K) !shear stress used to erode a portion of the core
    READ (10, *) (EORATE(CORE, LL, K), LL=1, KB) !erosion rate for each layer
                                                subject to shear stress TAULOC
  ENDDO
ENDDO

```

`s_sedic.f90` sets the EFDC variables equal to the information loaded from the input files. The critical shear stress of the initial bed, TAUCOR (dynes/cm²), and ERATE (equivalent of EORATE; cm/s) are set equal to the corresponding values from `erate.sdf` with only a minor change in the matrix indices to correlate with the convention of EFDC. The weight fraction, PER, is calculated by dividing the particle's mass percentage PNEW by 100, converting mass percentage to mass fraction. Also, the bulk density (BULKDENS; g/cm³) is the mass of solids per unit volume:

$$\chi_s \rho_s = \frac{\rho_s (\rho - 1)}{\rho_s - \rho_w}, \quad (1)$$

where χ_s is the mass fraction (PER) of the sediment in the bed, ρ_s is the density of the sediment (SEDDENS), the product ($\chi_s \rho_s$) is the mass of solids per unit volume (BULKDENS), ρ_w is the density of water (WATERDENS), and ρ is the density of the sediment bed (BDEN). The sediment bed density (BDEN) is defined as:

$$\rho = \chi_s \rho_s + \chi_w \rho_w, \quad (2)$$

where the mass fractions χ_s and χ_w sum to 1.

```

DO L=2, LA
  I=IL(L) !I location as a function of L
  J=JL(L) !J location as a function of L
  TAUCOR(:,L)=TAUTEMP(NCORENO(I,J),:) !Critical shear stresses from cores
  DO LL=1, KB
    DO M=1, ITBM
      ERATE(LL,L,M)=EORATE(NCORENO(I,J),LL,M) !set erosion rate to
measured value
    ENDDO
  ENDDO
  DO LL=1, KB
    DO K=1, NSCM
      PER(K,LL,L)=PNEW(NCORENO(I,J),LL,K)/100.0 !set mass fraction to
measured value
    ENDDO
    BULKDENS(LL,L)=((SEDDENS)/(SEDDENS-WATERDENS))
      *(BDEN(NCORENO(I,J),LL)-1)
    !set bulk density equal to that from SEDflume data
  ENDDO
ENDDO

```

Once the bulk density (BULKDENS) and the erosion rate (ERATE) are updated, `s_sedic.f90` reads in the layer thicknesses (TSED0S; g/cm^2). Note that the layer thicknesses are in units of cm from the `erate.sdf` file, but these quantities are converted later for ease of mass flux calculations. The bed layer thicknesses (i.e., BEDLINIT and HBED; m) are loaded and modified so that the bottom three layers always have a thickness of at least 10^{-12} and the EFDC variable HBED is equated to these data. The bed initial thickness BEDLINIT is reported in units of meters, hence the thickness values read in from the input file TSED0S is multiplied by 0.01.

```

READ (10, '(A80)') STR_LINE
READ (10, *) (TSED0S(LL), LL=1, KB)
FORALL (LL=3:KB) BEDLINIT(2:LA, LL)=0.01*MAX(1E-12, TSED0S(LL))
FORALL (LL=1:KB) HBED(2:LA, LL)=BEDLINIT(2:LA, LL)

```

`s_sedic.f90` activates any layers (i.e., LAYER = 1) with thickness greater than zero (TSED0S > 0). The layer's thickness (TSED and TSED0; g/cm^2) are then defined with layer thickness (TSED0S; cm) and the mass per unit volume (BULKDENS; g/cm^3).

```

FORALL (L=2:LA)
  WHERE (TSED0S(1:KB)>0.0)
    LAYER(1:KB,L)=1
  ELSEWHERE
    LAYER(1:KB,L)=0
  ENDWHERE
  FORALL (LL=1:KB)
    TSED(LL,L)=TSED0S(LL)*BULKDENS(LL,L)!TSED in units of  $\text{g}/\text{cm}^2$ .
    TSED0(LL,L)=TSED0S(LL)*BULKDENS(LL,L)!TSED0 in units of  $\text{g}/\text{cm}^2$ .
  ENDFORALL
ENDFORALL

```

The total sediment bed thickness (HBEDA; cm) is defined next. The average bed elevation (ZELBEDA; cm) is calculated with the bed elevation (BELV; cm) and the total sediment thickness (HBEDA; cm).

```
FORALL (L=2:LA) HBEDA(L) = SUM(BEDLINIT(L,1:KB))
FORALL (L=2:LA) ZELBEDA(L) = BELV(L) - HBEDA(L)
```

s_sedic.f90 re-reads bed.sdf to obtain information about newly deposited sediments. SCLOC is a vector of the sediment interpolant size classes used to interpolate erosion rates (ENRATE; cm/s) of newly deposited sediments. The critical shear stress for each sediment interpolant size class (SCLOC) is read in as TAUCRITE. The re-deposited erosion rate (ENRATE) value for each sediment interpolant size class (SCLOC) at different shear stresses (TAULOC) is read in next.

```
READ (30,'(A80)') STR_LINE
READ (30,*) (SCLOC(NSC),NSC=1,NSICM)
READ (30,'(A80)') STR_LINE
READ (30,*) (TAUCRITE(NSC),NSC=1,NSICM)
READ (30,'(A80)') STR_LINE
DO NSC=1,NSICM
  READ (30,*) (ENRATE(NSC,M),M=1,ITBM)
ENDDO
```

The settling velocity (cm/s) is (Cheng 1997):

$$w_s = \frac{\nu}{d} \left(\sqrt{25 + 1.2d^{*2}} - 5 \right)^{1.5}, \quad (3)$$

where w_s is the settling velocity, ν is the kinematic fluid viscosity (cm²/s), d is the particle diameter (cm), and d^* is the non-dimensional particle diameter defined as

$$d^* = d \left[\frac{(s_s - 1)g}{\nu^2} \right]^{1/3}, \quad (4)$$

where s_s is the sediment specific gravity (ρ_s/ρ_w), g is gravity (cm/s²), ν is the kinematic viscosity. All of these values are calculated in cgs units. (cm²/s) (van Rijn 1984a). In the code, DWS is the settling speed (cm/s) and the non-dimensional diameter (d^*) is DISTAR. Because d (D50) is reported in units of μm (in the code), it is divided by 10,000 to convert it to units of centimeters. The sediment specific of gravity, s_s , is approximately 2.65, gravity is 980.0 cm/s², and the kinematic viscosity of water is 0.01 cm²/s.

```
FORALL (K=1:NSCM)
  DISTAR(K) = D50(K) / 10000.0 * (1.65 * 980.0 / 0.01 ** 2) ** (1.0 / 3.0)
  DWS(K) = 0.01 / (D50(K) * 0.0001) * (SQRT(25.0 + 1.2 * DISTAR(K) ** 2) - 5.0) ** 1.5
ENDFORALL
```

s_sedic.f90 also calls on two input files (fetch.inp and stwave.inp) to read information about wind and wave forcing functions.


```

IF (IFWAVE.EQ.1) THEN
  OPEN(UNIT=50,FILE='fetch.inp')
  DO L=2,LA
    READ (50,*) I,J,(FWDIR(LIJ(I,J),FDIR),FDIR=1,8)
  ENDDO
  CLOSE(50)
ELSEIF (IFWAVE.EQ.2) THEN
  OPEN(UNIT=51,FILE='stwave.inp')
  DO NSKIP=1,5
    READ (51,'(A80)') STR_LINE
  ENDDO
  READ(51,*) STWVNUM,STWVTIM
  DO NWV=1,STWVNUM
    READ (51,'(A80)') STR_LINE
    DO L=2,LA
      READ(51,*) IWV,JWV,STWVHTMP,STWVTTMP,STWVDTMP
      STWVHT(LIJ(IWV,JWV),NWV)=STWVHTMP
      STWVTP(LIJ(IWV,JWV),NWV)=STWVTTMP
      STWVDR(LIJ(IWV,JWV),NWV)=STWVDTMP
    ENDDO
    !Incremental counter for which wave data set we are on
    STINC=0
    NWVCOUNT=STWVTIM-1
  ENDDO
  CLOSE(51)
ENDIF

```

The routine ends after printing to the screen and closing the input files.

SCANSEDZLJ.f90

SCANSEDZLJ.f90 loads data from bed.sdf and evaluates its compatibility with parameters such as number of sediment layers, particle size class, etc. from EFDC.INP. The routine identifies errors in the input files.

```
WRITE(*,'(A)') 'SCANNING INPUT FILE: BED.SDF'  
OPEN(1,FILE='BED.SDF',STATUS='OLD')  
READ(1,*,IOSTAT=ERROR) !SKIP THIS LINE  
IF(ERROR==1) THEN  
    WRITE(*,('READ ERROR IN SEDZLJ INPUT FILE'))  
    WRITE(8,('READ ERROR IN SEDZLJ INPUT FILE'))  
    STOP  
ENDIF  
READ(1,*,IOSTAT=ERROR) IDUMMY, IDUMMY, IDUMMY, KB  
IF(ERROR==1) THEN  
    WRITE(*,('READ ERROR IN SEDZLJ INPUT FILE'))  
    WRITE(8,('READ ERROR IN SEDZLJ INPUT FILE'))  
    STOP  
ENDIF  
READ(1,*,IOSTAT=ERROR) !SKIP THIS LINE  
IF(ERROR==1) THEN  
    WRITE(*,('READ ERROR IN SEDZLJ INPUT FILE'))  
    WRITE(8,('READ ERROR IN SEDZLJ INPUT FILE'))  
    STOP  
ENDIF  
READ(1,*,IOSTAT=ERROR) ITBM, NSICM  
IF(ERROR==1) THEN  
    WRITE(*,('READ ERROR IN SEDZLJ INPUT FILE'))  
    WRITE(8,('READ ERROR IN SEDZLJ INPUT FILE'))  
    STOP  
ENDIF
```

S MAIN.f90

This routine calculates sediment distributions in the various water layers. It is called from CALSED.for when EFDC calculates cohesive sediment transport. s_main.f90 replaces the current cohesive and non-cohesive sediment transport routine in EFDC. The first step in s_main.f90 involves calling the s_shear.f90 subroutine to calculate SEDZLJ shear stresses.

```
CALL SEDZLJ_SHEAR
```

If the morphology feature is activated, then s_main.f90 calculates sediment bed height (HBED; m) for each layer in each cell. The layer thickness, TSED (g/cm^2), is divided by the solid mass per volume (BULKDENS) to obtain HBED in units of length.

```
IF (IMORPH_SEDZLJ==1) FORALL (L=2:LA) HBED(L,1:KB)=0.01*(TSED(1:KB,L)/BULKDENS(1:KB,L))
```

The suspended sediment concentration (SED; kg/m^3) is also saved in another variable (SEDS; kg/m^3) for later calculations in vertical turbulent dispersion and settling.

```
SEDS(2:LA,1:KC,1:NSCM)=SED(2:LA,1:KC,1:NSCM)
```

Next, s_main.f90 checks if bedload sediment transport feature (NCALC_BL) is active. ISEDTIME serves as a flag for the bedload sediment transport feature. The velocity profile in the model is allowed to develop in ISEDTIME number of time steps before the sediment transport is activated. If bedload sediment feature is activated (i.e., NCALC_BL = 1), s_main.f90 calls the bedload subroutine (BEDLOADJ). This subroutine evaluates fraction of sediment transported in bedload and in suspended load. BEDLOADJ also evaluates the bedload concentrations and bedload fluxes to and from the bed.

```
IF (NCALC_BL==1 .AND. N>=ISEDTIME) CALL BEDLOADJ
```

s_sedzlj.f90 is activated in EFDC.inp with an IWRSP value of 98. IWRSP = 98 sets NSEDFLUME = 1 (option 2 is for SEDTOX) in INPUT.for. Here, both the NSEDFLUME and ISEDTIME are checked by s_main.f90 before sediment transport calculation is performed.

```
IF (NSEDFLUME==1 .AND. N>=ISEDTIME) THEN
```

s_main.f90 calculates vertical dispersion of suspended sediment according to the original formulation by Hamrick (1992). Based on the number of layers, s_main.f90 evaluates the vertical dispersion distinctly. "IF" conditional statements are added to account for cases with one or two layers because they do not involve full vertical dispersion. The sediment flux due to vertical dispersion is a function of vertical velocity (WVEL; m/s), settling velocity (WSETA; m/s), and suspended sediment concentration (SED).

```

KCNOT1LOOP: IF (KC .NE. 1) THEN
  DO NS=1, NSCM
    FORALL (L=2:LA) !K=KC
      WVEL(L)=DELT*HPI(L)*DZIC(KC)
      CLEFT(L)=1.0+WSETA(L,KC-1,NS)*WVEL(L)
      CRIGHT(L)=MAX(SED(L,KC,NS),0.0)
      SED(L,KC,NS)=CRIGHT(L)/CLEFT(L)
      SEDF(L,KC-1,NS)=-WSETA(L,KC-1,NS)*SED(L,KC,NS)
    ENDFORALL
  KCNOT2LOOP: IF( KC .NE. 2) THEN
    DO K=KC-1,2,-1
      FORALL (L=2:LA)
        WVEL(L)=DELT*HPI(L)*DZIC(K)
        CLEFT(L)=1.0+WSETA(L,K-1,NS)*WVEL(L)
        CRIGHT(L)=MAX(SED(L,K,NS),0.0)-SEDF(L,K,NS)*WVEL(L)
        SED(L,K,NS)=CRIGHT(L)/CLEFT(L)
        SEDF(L,K-1,NS)=-WSETA(L,K-1,NS)*SED(L,K,NS)
      ENDFORALL
    ENDDO
  ENDIF KCNOT2LOOP
ENDDO
ENDIF KCNOT1LOOP

```

Once the vertical dispersion of suspended sediments has been calculated, `s_main.f90` calls `SEDZLJ` to calculate sediment transport between the sediment bed and water column.

```
CALL SEDZLJ(L)
```

Once sediment transport is complete, the volumetric total sediment flux (Q_{SBDTOP} ; $\text{kg/m}^2\text{s}$) is calculated as a function of the inverse specific gravity (SSGI) and the sediment flux ($SEDF$; $\text{kg/m}^2\text{s}$), initial void ratio in each layer of the bed ($VDRBED$), and sediment void ratio ($VDRDEPO$).

```

QSBDTOP(L)=SUM(SSGI(1:NSCM)*SEDF(L,0,1:NSCM))
DO NS=1, NSCM
  QWBDTOP(L)=QWBDTOP(L)+SSGI(NS)*(VDRBED(L,KBT(L))*MAX(SEDF(L,0,NS),0.0)
    & +VDRDEPO(NS)*MIN(SEDF(L,0,NS),0.0))
ENDDO

```

Next, `s_main.f90` goes through a series of numerical computations that estimate the sediment flux between water layers. Vertical sediment diffusion is due to turbulent dispersion and settling.

```

DO NS=1,NSCM
  IF(KC>3) THEN
    IF(ISTOPT(6)==1) THEN
      DO K=1,KC-1
        FORALL(L=2:LA)
          CRNUM(L)=1.0+DELT*WSETA(L,K,NS)*HPI(L)*DZIC(K+1)
          GRADSED(L)=(SED(L,K+1,NS)-SED(L,K,NS))/(DZC(K+1)+DZC(K))
          SEDAVG(L)=0.5*(SED(L,K+1,NS)+SED(L,K,NS)+1.0E-16)
          WSETA(L,K,NS)=-CRNUM(L)*DZC(K+1)*WSETA(L,K,NS)*GRADSED(L)/SEDAVG(L)
        ENDFORALL
      ENDDO
      !TVAR1S=LOWER DIAGONAL
      TVAR1S(2:LA,1)=0.0
      DO K=2,KC
        TVAR1S(2:LA,K)=MIN(WSETA(2:LA,K-1,NS),0.0)
      ENDDO
      !TVAR1N=UPPER DIAGONAL
      TVAR1N(2:LA,KC)=0.0
      DO K=1,KC-1
        TVAR1N(2:LA,K)=-MAX(WSETA(2:LA,K,NS),0.0)
      ENDDO
      !TVAR1W=MAIN DIAGONAL
      TVAR1W(2:LA,1)=DELTI*DZC(1)*HP(2:LA)-MIN(WSETA(2:LA,1,NS),0.0)
      DO K=2,KC-1
        TVAR1W(2:LA,K)=DELTI*DZC(K)*HP(2:LA)+MAX(WSETA(2:LA,K-1,NS),0.0)-
MIN(WSETA(2:LA,K,NS),0.0)
      ENDDO
      TVAR1W(2:LA,KC)=DELTI*DZC(KC)*HP(2:LA)+MAX(WSETA(2:LA,KC-1,NS),0.0)
      !TVAR1E=RIGHT HAND SIDE
      DO K=1,KC
        TVAR1E(2:LA,K)=DELTI*DZC(KC)*HP(2:LA)*SED(2:LA,K,NS)
      ENDDO
      !TVAR3S=BET,TVAR2N=U,TVAR2S=GAM ARE WORKING ARRAYS
      TVAR3S(2:LA)=TVAR1W(2:LA,1)
      TVAR2N(2:LA,1)=TVAR1E(2:LA,1)/TVAR3S(2:LA)
      DO K=2,KC
        TVAR2S(2:LA,K)=TVAR1N(2:LA,K-1)/TVAR3S(2:LA)
        TVAR3S(2:LA)=TVAR1W(2:LA,K)-TVAR1S(2:LA,K)*TVAR2S(2:LA,K)
        TVAR2N(2:LA,K)=(TVAR1E(2:LA,K)-TVAR1S(2:LA,K)*
          TVAR2N(2:LA,K-1))/TVAR3S(2:LA)
      ENDDO
      DO K=KC-1,1,-1
        TVAR2N(2:LA,K)=TVAR2N(2:LA,K)-TVAR2S(2:LA,K+1)*TVAR2N(2:LA,K+1)
      ENDDO
      SED(2:LA,K,NS)=TVAR2N(2:LA,K)
    ENDIF
  ENDIF
ENDDO

```

Once `s_main.f90` recalculates the sediment concentration profile due to turbulent dispersion, the sediment flux ($\text{kg/m}^2\text{s}$) due to settling is calculated by starting with the top layer of water followed by fluxes to the lower water layers. The sediment flux (SEDF) is calculated as the product of the water layer's height ($\text{HP} \times \text{DZC}$; m), change in sediment concentration over the last time step ($\text{SED} - \text{SEDS}$), and the inverse time step, DELTI (1/s). For water layers others below

the top layer, the sediment flux (SEDF) is the sum of current layer's concentration and flux of the layer directly above it.

```

SEDF(2:LA,KS,NS)=DELTI*DZC(KC)*HP(2:LA)*
                (SED(2:LA,KC,NS)-SEDS(2:LA,KC,NS))
DO K=KS-1,1,-1 !KS=KC-1
    SEDF(2:LA,K,NS)=DELTI*DZC(K+1)*HP(2:LA)*(SED(2:LA,K+1,NS)-
        SEDS(2:LA,K+1,NS))+SEDF(2:LA,K+1,NS)
ENDDO

```

The sediment flux algorithm depends on the number of water layers and `s_main.f90` accounts for this with four distinct cases. The first case is for problems with more than 3 water layers, the second is for problems with exactly 3 water layers, the third case is for problems with 2 water layers, and the final case is for problems with only 1 water layer (the 2D and 1D problems).

Once the suspended sediment flux is calculated, `s_main.f90` calls the morphology routine `s_morph.f90` to update the grid for the ensuing hydrodynamics.

```

IF(IMORPH_SEDZLJ==1)CALL MORPHJ

```

S_SHEAR.f90

The shear subroutine uses the hydrodynamic results from EFDC to calculate the total shear stress at the sediment bed. This shear is used in the sediment erosion calculations. The shear stress and friction factor in `s_shear.f90` are calculated using Christofferson and Jonsson (CJ) log-law (1985), which is a function of the wind, wave, and convective information from the hydrodynamics of EFDC.

`s_shear.f90` first checks if the constant shear assumption is activated in the `bed.sdf` file.

```
IF (TAUCONST==0) THEN
```

If the TAUCONST flag is activated (i.e., not equal to zero), then `s_shear.f90` calculates the shear stress based on the local velocities of each cell, accounting for shear stress from the velocity gradient as well as wind- and wave-induced shear stresses. First, wave characteristics are calculated (i.e., wavelength, orbital velocity, height, and frequency) according to the lines of code below.

```
IF (IFWAVE==1) THEN
!Wind Wave Fetch
!Convert wind input into current wind info for wind-driven wave calcs
  IF (ISDYNSTP==0) THEN
    WFTIM=DT*FLOAT(N)/TCWSER(1)+TBEGIN*(TCON/TCWSER(1))
  ELSE
    WFTIM=TIMESEC/TCWSER(1)
  ENDIF
  M1=MWTLAST(1)
  M2=M1+1
  DO WHILE (TWSER(M2,1)<WFTIM)
    M1=M2
    M2=M1+1
  END DO
  MWTLAST(1)=M1
  TDIFF=TWSER(M2,1)-TWSER(M1,1)
  WTM1=(TWSER(M2,1)-WFTIM)/TDIFF
  WTM2=(WFTIM-TWSER(M1,1))/TDIFF
  FWINDS=WTM1*WINDS(M1,1)+WTM2*WINDS(M2,1)
  IF (FWINDS>1.0) THEN
    IF (ABS(WINDD(M1,1)-WINDD(M2,1))<180.0) THEN
      FWINDD=WTM1*WINDD(M1,1)+WTM2*WINDD(M2,1)
    ELSE
      IF (WINDD(M1,1).GT.WINDD(M2,1)) THEN
        FWINDD=WTM1*WINDD(M1,1)+WTM2*(WINDD(M2,1)+360.0)
      ELSE
        FWINDD=WTM1*(WINDD(M1,1)+360)+WTM2*WINDD(M2,1)
      ENDIF
      IF (FWINDD>=360.0) FWINDD=FWINDD-360.0
    ENDIF
    !Convert wind into direction it is blowing "from"
    IF (FWINDD<=180.0) THEN
      FWINDD=FWINDD+180.0
      IF (FWINDD==360.0) FWINDD=0.0
    ELSE

```

```

        FWINDD=FWINDD-180.0
        IF (FWINDD==360.0)FWINDD=0.0
ENDIF
!Calculate which of the 8 wind zones (FWZONE) the wind is coming from
!Also the Waveangle CCW from East
IF (FWINDD>=337.5.OR.FWINDD<22.5) THEN
    FZONE=1
    WVANGLE=4.712
ELSEIF (FWINDD>=22.5.AND.FWINDD<67.5) THEN
    FZONE=2
    WVANGLE=3.927
ELSEIF (FWINDD>=67.5.AND.FWINDD<112.5) THEN
    FZONE=3
    WVANGLE=3.142
ELSEIF (FWINDD>=112.5.AND.FWINDD<157.5) THEN
    FZONE=4
    WVANGLE=2.356
ELSEIF (FWINDD>=157.5.AND.FWINDD<202.5) THEN
    FZONE=5
    WVANGLE=1.571
ELSEIF (FWINDD>=202.5.AND.FWINDD<247.5) THEN
    FZONE=6
    WVANGLE=0.7854
ELSEIF (FWINDD>=247.5.AND.FWINDD<292.5) THEN
    FZONE=7
    WVANGLE=0.
ELSEIF (FWINDD>=292.5.AND.FWINDD<337.5) THEN
    FZONE=8
    WVANGLE=5.4978
ENDIF
!Calculate Domain Average Depth
!Needs to be calculated along each fetch
!This is sufficient for small systems
AVGDEPTH=SUM(HP(2:LA))/FLOAT(LA-1)
FWINDSQ=FWINDS*FWINDS
!Calculate wave height, period, orbital velocity, and length
DO L=2,LA
    FC1=(FWINDSQ/9.8)*0.283*TANH(0.530*(9.8*AVGDEPTH/FWINDSQ)**0.75)
    FC2=TANH(0.0125*(9.8*FWDIR(L,FZONE)/FWINDSQ)**0.42/
        TANH(0.530*(9.8*AVGDEPTH/FWINDSQ)**0.75))
    FWVHT(L)=MIN(HP(L),FC1*FC2)
    FC1=(FWINDS/9.8)*7.54*TANH(0.833*(9.8*AVGDEPTH/FWINDSQ)**0.375)
    FC2=TANH(0.077*(9.8*FWDIR(L,FZONE)/FWINDSQ)**0.25/
        TANH(0.833*(9.8*AVGDEPTH/FWINDSQ)**0.375))
    FWVTP(L)=FC1*FC2
    WVFREQ(L)=2.0*PI/FWVTP(L)
    FC1=(2.0*PI/FWVTP(L))**2*HP(L)/9.8
    FC2=FC1+1.0/(1.0+0.6522*(FC1)+0.4622*(FC1)**2+
        0.0864*(FC1)**4+0.0675*(FC1)**5)
    WVLENGTH=FWVTP(L)*SQRT(9.8*HP(L)/FC2)
    WVORBIT(L)=MAX(0.01,PI*FWVHT(L)/(FWVTP(L)*SINH(HP(L)*2.0*PI/WVLENGTH))
ENDDO
!*****
!Read in EFDC STWAVE Wave Field
ELSEIF (IFWAVE==2) THEN
    NWVCOUNT=NWVCOUNT+1
IF (NWVCOUNT==STWVTIM) THEN

```



```

NWVCOUNT=0
STINC=STINC+1
DO L=2, LA
  IF (STINC>STWVNUM) EXIT
  IF (STWVTP(L, STINC)>0.0) THEN
    WVFREQ(L)=2.0*PI/STWVTP(L, STINC)
    FWVHT(L)=MAX(HP(L), FWVHT(L))
    FC1=(2.0*PI/STWVTP(L, STINC))**2*HP(L)/9.8
    FC2=FC1+1.0/(1.0+0.6522*(FC1)+0.4622*(FC1)**2+0.0864*(FC1)**4+
      0.0675*(FC1)**5)
    WVLENGTH=STWVTP(L, STINC)*SQRT(9.8*HP(L)/FC2)
    WVORBIT(L)=MAX(0.01, PI*STWVHT(L, STINC)/(STWVTP(L, STINC)*
  SINH(HP(L)*2.0*PI/WVLENGTH))
  WVANG(L)=STWVDR(L, STINC)
  ELSE
    WVFREQ(L)=0.0
    WVORBIT(L)=0.0
    WVANG(L)=0.0
  ENDIF
ENDDO
ENDIF
ENDIF
ENDIF

```

After the wave characteristics are calculated, `s_shear.f90` can calculate shear stresses. The bed-form roughness is determined in three distinct ways. If the bed-form roughness is not provided from the `SEDZLJ` subroutines (i.e., `ZBSKIN = 0` and `NSEDFLUME = 0`), it is set equal to the `EFDC` roughness (`ZBR`, as loaded from `EFDC.INP`). If `NSEDFLUME` is greater than 0 but no bed-form roughness (`ZBSKIN`) is specified, then the bed-form roughness is set equal to the grain roughness (`D50AVE`). Otherwise, the bed-form roughness is equal to the value specified in `bed.sdf` (`ZBSKIN`).

```

DO L=2, LA
  ZBTEMP(L)=ZBR(L)
  IF (ZBSKIN.EQ.0.AND.NSEDFLUME.GT.0) THEN
    IF (D50AVG(L).LT.D50(1)) THEN
      ZBTEMP(L)=D50(1)/1.e6
    ELSE
      ZBTEMP(L)=D50AVG(L)/1.e6
    ENDIF
  ELSEIF (ZBSKIN.GT.0.AND.NSEDFLUME.GT.0) THEN
    ZBTEMP(L)=ZBSKIN/1.e6
  ENDIF
  KN(L)=30.0*ZBTEMP(L)
ENDDO

```

To obtain the shear stress, `s_shear.f90` calculates the average velocity magnitude.

```

DO L=2, LA
!Calculate Average Velocity Magnitude in cm/s
UTMP=100.0*STCUV(L)*(UHE(L+1)+UHE(L))/(HU(L+1)+HU(L))+1.0E-12
VTMP=100.0*STCUV(L)*(VHE(LNC(L))+VHE(L))/(HV(LNC(L))+HV(L))
VELMAG=SQRT(UTMP**2+VTMP**2)

```

The friction factor (FC) is then calculated using the water height and bed-form roughness (ZBTEMP). The friction factor is calculated with the CJ log-law.

$$FC(L) = (0.42 / \text{LOG}(HP(L) / (2.0 * ZBTEMP(L)))) ** 2$$

If there are no wave and wind inputs, then the shear stress can be calculated directly with only the velocity magnitude and the friction factor using the CJ log-law. Otherwise, a more complex method is used to incorporate the wind- and wave-induced shear into the shear (TAU; dynes/cm²) calculation.

```

IF (IFWAVE==0.AND.UWVSQ(L)==0.0.OR.WVORBIT(L)==0.0) THEN
TAU(L)=FC(L)*VELMAG**2
ELSE !Calculate Current Angle CCW From X axis
IF (UTMP>0.0.AND.VTMP>0.0) THEN
VELANG=ATAN(VTMP/UTMP)
ELSEIF (UTMP<0.0) THEN
VELANG=ATAN(VTMP/UTMP)+PI
ELSEIF (UTMP>0.0.AND.VTMP<0.0) THEN
VELANG=2*PI+ATAN(VTMP/UTMP)
ELSEIF (UTMP==0.0) THEN
VELANG=SIGN(0.5*PI,VTMP)
ENDIF
IF (IFWAVE==0) THEN !Set Orbital velocity in m/s and waveangle and
frequency
WVORBIT(L)=SQRT(UWVSQ(L))
WVFREQ(L)=WVFRQ
WVANG(L)=WACCWE(L)
ELSEIF (IFWAVE==1) THEN
WVANG(L)=WVANGLE
ENDIF
!Calculate wave friction factor
FWW(L)=2.0*(0.0747*(KN(L)*WVFREQ(L)/WVORBIT(L)))**0.66
SIGMAWV=FC(L)/FWW(L)*(VELMAG/(WVORBIT(L)*100.0))**2
MMW=SQRT(1.0+SIGMAWV**2+2.0*SIGMAWV*ABS(COS(VELANG-WVANG(L))))
JJW=WVORBIT(L)/(KN(L)*WVFREQ(L))*SQRT(MMW*FWW(L)/2.0)
FWW(L)=MMW*0.15/JJW
!Calculate wave boundary layer info
DELW=KN(L)*0.273*SQRT(JJW)
APROUGH=30.0*DELW*EXP(-5.62*DELW/KN(L)*SQRT(SIGMAWV/MMW))
!Calculate new current friction factor
FC(L)=2.0*(1.0/(2.38*LOG(30.0*HP(L)/(2.718*KN(L)))-
2.38*LOG(APROUGH/KN(L))))**2
!Iterate once more
SIGMAWV=FC(L)/FWW(L)*(VELMAG/(WVORBIT(L)*100.0))**2
MMW=SQRT(1.0+SIGMAWV**2+2.0*SIGMAWV*ABS(COS(VELANG-WVANG(L))))
JJW=WVORBIT(L)/(KN(L)*WVFREQ(L))*SQRT(MMW*FWW(L)/2.0)
FWW(L)=MMW*0.15/JJW

```

The total shear stress (TAU) is calculated in cgs units (dynes/cm²). This is converted to Pascals and renamed TAUB. With TAUB, the shear velocity is calculated as:

$$u^* = \sqrt{\frac{\tau}{\rho_w}}, \quad (5)$$

where u^* (USTAR) is the shear velocity (m/s), τ (TAUB) is the shear stress (Pa), and ρ_w (RHO; kg/m³) is the density of water (Jones 2001, page 14). Shear velocity is used in many of the subsequent calculations for sediment transport.

```
!Calculate total wave and current shear stress (dynes/cm^2)
  TAU(L)=0.5*FWW(L)*WVORBIT(L)**2*10000.0*MMW
  TAUB(L)=0.1*TAU(L)
  USTAR(L)=SQRT(TAUB(L)/RHO)
ENDIF
ENDDO
```

If the shear stress is defined to be constant in bed.sdf, then s_shear.f90 skips all of the calculation of friction- and wave-induced shear and these quantities are set equal to the constant shear value defined in the input file.

```
ELSE
  DO L=2, LA
    TAU(L)=TAUCONST
    TAUB(L)=0.1*TAU(L)
    USTAR(L)=SQRT(TAUB(L)/1000.)
  ENDDO
ENDIF
```

S BEDLOAD.f90

s_bedload.f90 first calculates the ratio of shear velocity to settling velocity. This ratio (USW) is a function of shear stress (TAU; dynes/cm²), density of water (WATERDENS), and settling velocity (DWS; cm/s).

$$USW(L, K) = \text{SQRT}(TAU(L) / WATERDENS) / DWS(K)$$

Next, the subroutine determines the fraction of sediment in suspended transport versus bedload transport using both flow and sediment bed properties. These include sediment characteristics, shear stress, and shear velocity at each cell. Using the empiricism of Guy et al. (1966), s_bedload.f90 assumes that all particles with a diameter less than 200 μm are transported as suspended load (unless there is no flow). Particles with an average diameter greater than 200 μm and a ratio of shear velocity (USTAR) to settling velocity (DWS) less than 4 are partially transported as bedload. If the shear stress is less than the critical shear stress, there is no erosion or suspension. The fraction of suspended sediment is calculated as (Jones 2001, eq. 3.7):

$$\frac{q_s}{q_t} = \begin{cases} 0 & \text{for } \tau < \tau_{cs} \\ \frac{\ln\left(\frac{u^*}{w_s}\right) - \ln\left(\frac{\sqrt{\tau_{cs}/\rho_w}}{w_s}\right)}{\ln(4) - \ln\left(\frac{\sqrt{\tau_{cs}/\rho_w}}{w_s}\right)} & \text{for } \tau > \tau_{cs} \text{ and } \frac{u^*}{w_s} < 4, \\ 1 & \text{for } \frac{u^*}{w_s} > 4 \end{cases} \quad (6)$$

where q_s/q_t is the ratio of suspended load sediment flux over total sediment flux, u^* is defined in (5), w_s is the settling velocity (cm/s), τ_{cs} (dynes/cm²) is the critical shear stress for suspension, and ρ_w (g/cm³) is the water density.

```
LCM_LOOP_1:DO L=2, LA
WHERE (D50(1:NSCM) >= 200.0 .AND. USW(L, 1:NSCM) < 4.0)
  WHERE (TAU(L) <= TCRE(1:NSCM) .OR. USW(L, 1:NSCM) <= 0.0)
    BLFLAG(L, 1:NSCM) = 0
    PSUS(L, 1:NSCM) = 0.0
  ELSEWHERE
    BLFLAG(L, 1:NSCM) = 1
    PSUS(L, 1:NSCM) = MAX((LOG(USW(L, 1:NSCM)) -
      LOG(SQRT(TCRSUS(1:NSCM))/DWS(1:NSCM)))/(LOG(4.0) -
      LOG(SQRT(TCRSUS(1:NSCM))/DWS(1:NSCM))), 0.0)
  ENDWHERE
ELSEWHERE
  BLFLAG(L, 1:NSCM) = 0
  PSUS(L, 1:NSCM) = 1.0
ENDWHERE
ENDDO LCM_LOOP_1
```

After evaluating the fraction of sediment in bedload and suspension, `s_bedload.f90` calculates the bedload velocity (BLVEL; cm/s). First, the velocity magnitude (VELMAG; m/s) is calculated in MKS units. A dimensionless transport parameter (TRANS), is defined as a function of the shear stress (TAU; dynes/cm²) and the critical shear stress (TCRE; dynes/cm²) provided in the `bed.sdf` input file, is calculated (van Rijn 1984a).

$$\tau_* = \frac{\tau - \tau_{ce}}{\tau_{ce}}, \quad (7)$$

where τ_* is the transport parameter, τ (dynes/cm²) is shear stress at bed surface, and τ_{ce} (dynes/cm²) is critical shear stress for erosion.

Next, `s_bedload.f90` calculates the saltation height, DZBL (cm), which is a function of transport parameter (TRANS), particle size class (D50), and non-dimensional particle diameter (DSTAR) calculated in `s_sedic.f90` (van Rijn 1984a).

$$h_b = 0.3dd^{*0.7}\tau_*^{0.5}, \quad (8)$$

where h_b (cm) is the saltation height.

The bedload velocity u_b (BLVEL; cm/s) is calculated using the transport parameter (TRANS) and the size class (D50).

$$u_b = 1.5\tau_*^{3/5} \left[\left(\frac{\rho_s}{\rho_w} - 1 \right) gd \right]^{1/2}. \quad (9)$$

```
LCM_LOOP_2:DO L=2,LA
  VELMAG(L) =SQRT(U(L,1)**2+V(L,1)**2)
  TRANS(L,K)=MAX((TAU(L)-TCRE(K))/TCRE(K),0.0)
  DZBL(L,K)=MIN(100.*HP(L),
                D50(K)/10000.0*0.3*DSTAR(K)**0.7*SQRT(TRANS(L,K)))
  BLVEL(L,K)=1.5*TRANS(L,K)**0.6*SQRT((SSG-1)*980.0*D50(K)/10000.0)
ENDDO LCM_LOOP_2
```

After the bedload velocity magnitude (BLVEL; cm/s) is calculated, `s_bedload.f90` converts it into two orthogonal components. The equation below involves the ratios of U and VELMAG and V and VELMAG, (the x- and y-components of the hydrodynamic velocity and the velocity magnitude; m/s), respectively. Therefore the units of VELMAG, U, and V are irrelevant. Ultimately, the x- and y-components of bedload velocity (UBL and VBL; cm/s) are in cgs units.

```
WHERE(VELMAG(2:LA)>0.0)
  UBL(2:LA,K)=BLVEL(2:LA,K)*U(2:LA,1)/VELMAG(2:LA)
  VBL(2:LA,K)=BLVEL(2:LA,K)*V(2:LA,1)/VELMAG(2:LA)
ELSEWHERE
  UBL(2:LA,K)=0.0
  VBL(2:LA,K)=0.0
ENDWHERE
```

Next, `s_bedload.f90` calculates the bedload sediment flux. Three loops in `s_bedload.f90` solve the following differential equation (Jones 2001, eq. 3.23):

$$\frac{\partial(h_b C_b)}{\partial t} = \frac{\partial q_{b,x}}{\partial x} + \frac{\partial q_{b,y}}{\partial y} + Q_b, \quad (10)$$

where C_b (g/cm^3) is the bedload concentration, $q_{b,x}$ (g/cms) is the bedload sediment line flux in the x direction, $q_{b,y}$ (g/cms) is the bedload sediment line flux in the y direction, and Q_b ($\text{g/cm}^2\text{s}$) is the sediment flux between the sediment bed and the bedload. For example, $q_{b,x} = u_{b,x} h_b C_b$, where $u_{b,x}$ is the x -component (analogously for y) of the bedload velocity from Eq. (9). The component ratio of $u_{b,x}$ to $u_{b,y}$ is equal to the ratio of U to V from the EFDC hydrodynamics routine.

The first loop solves for $q_{b,x} dt$ (XBLFLUX; g/cm^3) in discretized form.

```
LCM_LOOP_3:DO L=2,LA
  I=IL(L)
  J=JL(L)
  IF(LIJ(I-1,J)>0) THEN
    IF(UBL(L,K).GT.0) THEN
      XBLFLUX(L,K)=DT/(DXP(L)*100.0)*CBL(1,LIJ(I-1,J),K)*UBL(LIJ(I-1,J),K)*
        DZBL(LIJ(I-1,J),K)
    ELSE
      XBLFLUX(L,K)=DT/(DXP(L)*100.0)*CBL(1,LIJ(I,J),K)*UBL(LIJ(I,J),K)*
        DZBL(LIJ(I,J),K)
    ENDIF
  ELSE
    XBLFLUX(L,K)=0.0
  ENDIF
ENDDO LCM_LOOP_3
```

The second loop solves for $q_{b,y} dt$ (YBLFLUX; g/cm^3) in discretized form.

```
LCM_LOOP_5:DO L=2,LA
  I=IL(L)
  J=JL(L)
  IF(VBL(L,K).GE.0) THEN
    YBLFLUX(L,K)=DT/(DYP(L)*100.0)*CBL(1,LSC(L),K)*VBL(LSC(L),K)*
      DZBL(LSC(L),K)
  ELSE
    YBLFLUX(L,K)=DT/(DYP(L)*100.0)*CBL(1,LIJ(I,J),K)*VBL(LIJ(I,J),K)*
      DZBL(LIJ(I,J),K)
  ENDIF
ENDDO LCM_LOOP_5
```

After the first two terms on the right hand side of the equation are evaluated, the bedload sediment concentration (CBL; g/cm³) is calculated.

```

LCM_LOOP_4:DO L=2,LA
  I=IL(L)
  J=JL(L)
  IF(BLFLAG(L,K)==1) THEN
    IF(LIJ(I+1,J)>0) THEN
      CBL(2,L,K)=CBL(1,L,K)+(XBLFLUX(L,K)-XBLFLUX(LIJ(I+1,J),K)+YBLFLUX(L,K)-
        YBLFLUX(LNC(L),K)+QBSED(L,K))/DZBL(L,K)
    ELSE
      CBL(2,L,K)=CBL(1,L,K)+(YBLFLUX(L,K)-
        YBLFLUX(LNC(L),K)+QBSED(L,K))/DZBL(L,K)
    ENDIF
  ELSE
    CBL(2,L,K)=0.0
  ENDIF
  IF(DZBL(L,K)==0.OR.CBL(2,L,K)<0)CBL(2,L,K)=0.0
ENDDO LCM_LOOP_4
ENDDO SED_LOOP
CBL(1,2:LA,1:NSCM)=CBL(2,2:LA,1:NSCM)

```

S SEDZLJ.f90

The `s_sedzlj.f90` routine calculates erosion and deposition of the active layer (top layer of the bed) for all cells. First, the following variables are initialized: total erosion from each cell (ETOTO; g/cm^2), total deposition of each cell (DEP; g/cm^2), water column depth (HP; m) is converted to cm and renamed (HT; cm), suspended sediment deposition (D; g/cm^2), bedload deposition (DBL; g/cm^2), total erosion of each cell (E; g/cm^2), total erosion of the top layer (ELAY; g/cm^2), and sediment mass available for deposition (SMASS; g/cm^2).

```
ETOTO(L)=0.0
DEP(L)=0.0
HT(L)=HP(L)*100.0
D(1:NSCM,L)=0.0
DBL(1:NSCM,L)=0.0
E(1:NSCM,L)=0.0
ELAY(1:NSCM)=0.0
SMASS(1:NSCM)=0.0
```

After initializing erosion and deposition variables, `s_sedzlj.f90` calculates the near-bed concentration (CTB; g/cm^3). If there is only one layer in the water column, the near-bed concentration is calculated with the approximation that the suspended sediment concentration has the following profile (Jones 2001, eq. 3.12):

$$C_{s,k}(z) = C_0 e^{-\frac{w_{s,k}z}{D_v}}, \quad (11)$$

where $C_{s,k}(z)$ (g/cm^3) is the suspended concentration as a function of water depth, C_0 (g/cm^3) is the near-bed concentration, $w_{s,k}$ is the settling speed, and D_v is the vertical dispersivity. For single water layer systems, the vertical dispersivity (D_v) is calculated from an empirical relation for rivers (Fischer et al. 1979) as:

$$D_v = 0.067 hu^*, \quad (12)$$

where D_v (cm^2/s) and h (cm) is the water depth. In the code below, VZDIF (cm^2/s) is the vertical dispersivity, HT (cm) is h , and USTAR (cm/s) is u^* . A lower limit of $20.0 \text{ cm}^2/\text{s}$ is set for the vertical dispersivity. The average suspended sediment concentration (C_{ave} or SED; g/cm^3) is assumed to be the vertically-integrated average concentration. With the vertical diffusion, `s_sedzlj.f90` calculates the near-bed concentration C_0 (CTB in the code). C_0 is solved according to

$$C_{\text{ave}} = \int_0^h C_0 e^{-\frac{w_{s,k}z}{D_v}} dz \Rightarrow C_0 = C_{\text{ave}} \left(-\frac{w_{s,k}h}{D_v} \right) \left(\frac{1}{1 - e^{-\frac{w_{s,k}h}{D_v}}} \right). \quad (13)$$

If there is more than one water layer in the water column, the near-bed concentration is assumed to be the cell-centered average sediment concentration of the first water layer (SED(L,1,N); g/cm^3).

```
IF (KC == 1) THEN
  DO K=1, NSCM
    VZDIF(L) = MAX(20.0, 0.067*HT(L)*USTAR(L)*100.)
```



```

TEMP2=HT(L)*DWS(K)/VZDIF(L)
CTB(K,L)=SED(L,1,K)*TEMP2*(1.0/(1.0-EXP(-TEMP2)))*1.0E-06
ENDDO
ELSE
CTB(1:NSCM,L)=SED(L,1,1:NSCM)*1.0E-06
ENDIF

```

The temporary thickness (TTEMP; g/cm²) of each sediment size class in the top layer is calculated as the product of the percentage of each sediment size class in the top (active) layer (PER) and the thickness of the top bed layer (TSED).

```
TTEMP(1:NSCM,L)=PER(1:NSCM,1,L)*TSED(1,L)
```

With the temporary thickness, `s_sedzlj.f90` calculates deposition from suspended load, first checking if the near-bed concentration for a particular size class is non-zero, i.e., $CTB(K,L) < 1.0E-10$. The probability of deposition from suspended load is a function of shear stress (TAU), critical shear stress (TCRSUS) from the SEDflume input files, and probability calculations from Gessler (1967) and Krone (1962).

The probability of deposition (P_k) of cohesive, sub-200- μ m, suspended sediment particles is (Krone 1962)

$$P_k = \begin{cases} 0 & \text{for } \tau > \tau_{cs} \\ 1 - \frac{\tau}{\tau_{cs}} & \text{for } \tau \leq \tau_{cs} \end{cases} \quad (14)$$

For sediment particles with a diameter greater than 200 μ m (non-cohesive), the probability of deposition is assumed to follow a Gaussian probability density function (Gessler 1967)

$$P_k = \text{erf}\left(\frac{Y}{2}\right), \quad (15)$$

where

$$Y = \frac{1}{\sigma} \left(\frac{\tau_{cs,k}}{\tau} - 1 \right). \quad (16)$$

The error function is approximated as (Abramowitz and Stegun 1972):

$$P_k \approx 1 - \exp\left[\left(-Y/2\right)^2\right] \left(0.34802 X - 0.09587 X^2 + 0.74785 X^3\right), \quad (17)$$

where $X = (1+0.3327Y)^{-1}$. When $Y < 0$,

$$P_k = 1 - P(|Y|). \quad (18)$$

In the lines of code below, P_k is PROB(K).

```

DO K=1,NSCM
IF(CTB(K,L)<1.0E-10)CYCLE
IF(D50(K)>=200.0)THEN
PY(K)=ABS(1.7544*(TCRSUS(K)/TAU(L)-1.0))
PFY(K)=0.39894*EXP(-0.5*PY(K)*PY(K))
PX(K)=1.0/(1.0+0.3327*PY(K))
PROB(K)=PFY(K)*(0.4362*PX(K)-0.1202*PX(K)*PX(K)+0.9373*PX(K)**3)

```

```

IF (TAU(L) <= TCRSUS(K)) PROB(K) = 1.0 - PROB(K)
ELSEIF (TAU(L) <= TCRSUS(K)) THEN
    PROB(K) = 1.0 - TAU(L) / (TCRSUS(K))
ELSE
    PROB(K) = 0.0
ENDIF

```

All sediment is deposited from the bottom water layer. DZC is the water-layer thickness weight-factor. The product of DZC and HT is the thickness of the each water layer in cm. This product is multiplied with the sediment concentration in the bottom water layer (SED(K,1,L)) to find the sediment mass available in each cell per unit area (SMASS). MAXDEPLIMIT is the maximum fraction of sediment from the bottom water layer allowed to deposit onto the bed per time step. This fraction is specified in `bed.sdf` and is usually set to 1.0 unless instability is observed.

```
SMASS(K) = SED(K,1,L) * DZC(1) * HT(L) * MAXDEPLIMIT
```

Suspended sediment deposition, D , is calculated as the product of probability of deposition (PROB), near-bed concentration (CTB), settling speed (DWS) and the time step (DT; s). Once the suspended sediment deposition is calculated, `s_sedzlj.f90` ensures that the deposited material is less than or equal to the total mass in the first water layer.

```

D(K,L) = PROB(K) * (DWS(K) * DT) * CTB(K,L)
ENDDO
D(1:NSCM,L) = MIN(MAX(D(1:NSCM,L), 0.0), SMASS(1:NSCM))

```

Next, `s_sedzlj.f90` calculates bedload deposition fraction. The conditions for bedload deposition to occur are:

1. Bedload concentration (CBL) must exist for a particular size class,
2. Bedload velocity (BLVEL) must be greater than zero, and
3. Bedload transport flag (BLFLAG) must be turned on.

The bedload equilibrium concentration is calculated according to Van Rijn (1984a, eqn. 21):

$$C_e = 0.18 \frac{C_0 \tau_*}{d_*}, \quad (19)$$

where C_e is the equilibrium concentration (CSEDVR; g/cm^3), C_0 is the maximum concentration (empirically calculated to be 0.65), τ_* is the transport parameter (TRANS) calculated in (7), and d_* is the dimensionless particle diameter from (4). In sediment transport, deposition is the product of probability of deposition, the settling velocity, and the concentration at the bed:

$$D = P w_s C, \quad (20)$$

where D ($\text{g/cm}^2\text{s}$) is the deposition rate and P is the probability of deposition (in the case of dynamic equilibrium P is also the probability of erosion).

In dynamic equilibrium where erosion and deposition rates are the same,

$$E = P w_s C_e, \quad (21)$$

where E ($\text{g/cm}^2\text{s}$) is the erosion rate.

From the equilibrium concentration, the probability of deposition is calculated as:

$$P = \frac{E_b}{w_s C_e}, \quad (22)$$

where P is the Van Rijn probability of deposition (PROBVR), E_b is the bedload erosion (EBL; g/cm^2), w_s is the settling velocity (DWS), and C_e is the equilibrium concentration (CSEDVR). Note that in the equation below for the amount of sediment eroded into bedload (CSEDSS; g/cm^3), DWS is multiplied by the time step DT to calculate the amount of sediment eroded at each time step. The value of EBL is the total mass eroded for that particular time step. Hence, to obtain the erosion rate E_b , EBL must be divided by DT.

If there is no equilibrium concentration (i.e., CSEDVR \leq 0.0), then all sediment in bedload deposits onto the bed (PROBVR = 1.0). If an equilibrium concentration exists (i.e., CSEDVR > 0), then the probability of deposition is the fraction of sediment eroded into bedload over the equilibrium concentration of bedload sediment (i.e., CSEDSS/CSEDVR).

```
DO K=1, NSCM
  IF (CBL(1, L, K) > 1.0E-10 .AND. BLVEL(L, K) > 0.0 .AND. BLFLAG(L, K) == 1) THEN
    CSEDSS(K) = EBL(L) / (DWS(K) * DT)
    CSEDVR(K, L) = (0.18 * 0.65 * TRANS(L, K) / DISTAR(K))
    IF (CSEDVR(K, L) <= 0.0) THEN
      PROBVR(K, L) = 1.0
    ELSE
      PROBVR(K, L) = MIN(CSEDSS(K) / CSEDVR(K, L), 1.0)
    ENDIF
  ENDIF
```

In the case when the sediment eroded into bedload is less than zero (CSEDSS < 0.0), probability of deposition reverts back to Gessler's formulation (1967) (i.e., PROBVR = PROB). This simulates the case when there is deposition to the bed layer and no erosion.

```
IF (CSEDSS(K) <= 0.0) PROBVR(K, L) = PROB(K)
```

Once the probability of bedload deposition is calculated, `s_sedzlj.f90` calculates the local bedload mass available for deposition, SMASS, which is a function of bedload concentration (CBL) and saltation height (DZBL; cm). Bedload sediment deposition (DBL) is a function of probability of bedload deposition (PROBVR), bedload concentration (CBL), settling speed (DWS), and time step (DT). The code ensures that bedload sediment deposited onto the bed does not exceed total mass in the bedload (SMASS).

```
SMASS(K) = CBL(1, L, K) * DZBL(L, K)
DBL(K, L) = PROBVR(K, L) * CBL(1, L, K) * (DWS(K) * DT)
ENDIF
ENDDO
DBL(1: NSCM, L) = MIN(MAX(DBL(1: NSCM, L), 0.0), SMASS(1: NSCM))
```

Total sediment deposited is the sum of deposition from the bedload (DBL) and suspended load (D). The deposition rate (DEPO; g/cm²s) results from dividing the total sediment deposited (DEP; g/cm²) by the time step (DT).

```
DEP(L)=SUM(DBL(1:NSCM,L),BLVEL(L,1:NSCM)>0.0)+SUM(D(1:NSCM,L))
DEPO(L)=DEP(L)/(DT)
```

Once deposition is calculated for each time step, erosion is calculated. First, local deposition is checked (DEP(L) > 0.0). Then top bed sediment layer thickness (TSED(1,L)) is modified according to the amount of deposited materials.

```
IF(DEP(L)>0.0) THEN
  LAYER(1,L)=1
  TSED(1,L)=TSED(1,L)+DEP(L)
```

If there is bedload transport (i.e., BLVEL > 0), the mass fraction (PER) is updated by dividing the sum of original mass in the first layer (TTEMP), mass deposited from the bedload (DBL), and suspended load (D) by the newly calculated top bed layer thickness (TSED). If there is no bedload transport, only the original mass and suspended load deposition are summed.

```
WHERE (BLVEL(L,1:NSCM)>0.0)
  PER(1:NSCM,1,L)=(TTEMP(1:NSCM,L)+DBL(1:NSCM,L)
    +D(1:NSCM,L))/(TSED(1,L))
ELSEWHERE
  PER(1:NSCM,1,L)=(TTEMP(1:NSCM,L)+D(1:NSCM,L))/(TSED(1,L))
ENDWHERE
ENDIF
```

In `s_sedic.f90`, at least three bed layers must exist to account for deposition and erosion calculations. Because one bed layers may erode entirely, the following code determines which bed layer is the active layer and which layer is below the active layer. Finding the bed layer directly below the active layer is necessary because there are cases where sediment bed layers erode away. If the top layer from the last time step is not the active layer (it has eroded away), then `s_sedzlj.f90` finds the active layer (SLLN = LL) and the layer below it. For example, if there are five bed layers, then the top two layers initially have zero thicknesses and they are always active (if there is deposition or armoring), otherwise their thickness remains zero. The third, fourth, and fifth layers will have an initial thickness determined by the original sediment bed. When there is deposition, the second layer will exist (i.e., LAYER(2,L)=1). If there is no deposition (i.e., DO LL=3,KB), the 'erodible' layer is the third layer. If all the sediment in the third layer has eroded away in a cell, the code will reassign the fourth layer as the 'erodible' layer in that cell.

```
IF(LAYER(2,L)==1) THEN
  SLLN(L)=2
ELSE
  DO LL=3,KB
    IF(LAYER(LL,L)==1.AND.LAYER(LL-1,L)/=1) THEN
      SLLN(L)=LL
      EXIT
    ENDIF
  ENDDO
ENDIF
```

Next, `s_sedzlj.f90` assigns the active layer (SURFACE).

```

IF(LAYER(1,L)==1)THEN !is the top layer present?
    SURFACE=1 !surface variable established
ELSE
    SURFACE=SLLN(L) !otherwise the top layer is SLLN
ENDIF

```

As a result of deposition and erosion of sediment at the active layer, the average particle size (D50AVE) of the top bed layer is updated with the particle size classes (D50) and the updated mass fractions (PER). The equivalent variable in EFDC, SEDDIA50, is also updated.

```

D50AVG(L)=SUM(PER(1:NSCM,SURFACE,L)*D50(1:NSCM))
FORALL(LL=1:KB) SEDDIA50(L,LL)=SUM(PER(1:NSCM,LL,L)*D50(1:NSCM))

```

If the bed surface is in the active layer (SURFACE = 1 or 2), then the critical shear stress of the bed is interpolated from the interpolant size classes (SCLOC) and the critical shear for the corresponding interpolant size class (TAUCRITE). For example, if the average diameter (D50AVE) falls between two interpolant size classes (SCLOC(1) and SCLOC(2), where TAUCRITE(2) > TAUCRITE(1)), then the critical shear stress (TAUCRIT) for the updated bed with an average particle size (D50AVE) is linearly interpolated using TAUCRITE(1) and TAUCRITE(2). In the case where the top layer is not layer 1 or 2 (the cell is purely erosive), critical shear is set equal to the critical shear in of the original bed (TAUCOR). There is no need for interpolation here because the critical shear stress data for the original bed (TAUCOR) is obtained via SEDflume.

```

DO K=1, NSICM-1
    IF(D50AVG(L)>=SCLOC(K).AND.D50AVG(L)<SCLOC(K+1))THEN
        NSCD(1)=SCLOC(K)
        NSCD(2)=SCLOC(K+1)
        NSC0=K
        NSC1=K+1
        EXIT
    END IF
ENDDO
IF(SURFACE==1.OR.SURFACE==2)THEN
    TAUCRIT(L)=TAUCRITE(NSC0)+(TAUCRITE(NSC1)-TAUCRITE(NSC0))/
        (NSCD(2)-NSCD(1))*(D50AVG(L)-NSCD(1))
ELSE
    TAUCRIT(L)=TAUCOR(SURFACE,L)
ENDIF

```

The active layer thickness is defined as the depth into the sediment bed (depth below water-sediment interface) that is influenced by the water shear stress. In the case of bed armoring, the larger sediment do not move while smaller particles within the interstices of the larger sediment are sheared away into the water column down to a certain thickness (the active layer thickness). The active layer is recalculated according to the bed armoring formula (Van Niekerk et al. 1992):

$$T_a = 2d_{50} \frac{\tau}{\tau_{ce}}, \quad (23)$$

where T_a (cm) is the active layer's thickness (TACT), d_{50} (μm) is the layer-averaged particle diameter (D50AVE; μm), τ (dynes/cm²) is the shear stress at the bed (TAU), and τ_{ce} (dynes/cm²) is the critical shear stress (TAUCRIT). (Note: the units of TACT in the code is (g/cm²) instead of cm like T_a). If the shear stress (TAU) is less than the critical shear stress (TAUCRIT), the active layer thickness (TACT) is proportional to the ratio of shear stress to critical shear stress (i.e., TAU/TAUCRIT). However, if the shear stress is greater than the critical shear stress, the active layer thickness (TACT) is only a function of average particle diameter (D50AVE). Note that the density (BULKDENS) exists in the equation for TACT because TACT has units of g/cm².

```

IF (TAU(L) / TAUCRIT(L) < 1.0) THEN
    TACT(L) = 2.0 * D50AVG(L) * (BULKDENS(1,L) / 10000.0)
ELSE
    TACT(L) = 2.0 * D50AVG(L) * (TAU(L) / TAUCRIT(L)) * (BULKDENS(1,L) / 10000.0)
ENDIF

```

s_sedzlj.f90 checks for the existence of an active layer, which happens when there is armoring or deposition, but not when the model cell is purely erosive.

```

NACTLAY=0
DO LL=1,KB
    IF (LAYER(LL,L) > 0) THEN
DO K=1,NSCM
    IF (PER(K,LL,L) > 0.0 .AND. TAU(L) < TCRE(K) .AND. TAU(L) > TAUCRIT(L)) THEN
        NACTLAY=1
        LAYER(1,L)=1
    ENDIF
ENDDO
EXIT
ENDIF
ENDDO

```

When material is deposited onto (or eroded away from) the top layer, the thickness of the top layer is adjusted back to the active layer thickness according to the following three scenarios:

1. If there is a net deposition and the top layer thickness TSED(1,L) is greater than the calculated active layer TACT(L), then mass from the top layer is transferred to the layer below it. The amount of mass transferred is TSED(1,L)–TACT(L). Once this is done, the mass fraction (PER) for both layers is recalculated.

```

IF (TSED(1,L) > 0.0 .OR. NACTLAY /= 0) THEN
    IF (LAYER(1,L) == 1) THEN
        IF (TSED(1,L) > TACT(L)) THEN
            FORALL (K=1:NSCM) PER(K,2,L) = (PER(K,2,L) * TSED(2,L)
                + PER(K,1,L) * (TSED(1,L)
                - TACT(L))) / (TSED(2,L)
                + (TSED(1,L) - TACT(L)))
            LAYER(2,L)=1
            SLLN(L)=2
            TSED(2,L)=TSED(2,L)+TSED(1,L)-TACT(L)
            TSED(1,L)=TACT(L)
        ENDIF
    ENDIF
ENDIF

```

2. When deposition occurs, but the net sediment flux is erosional, the calculated active layer thickness TACT(L) will be greater than the top layer thickness TSED(1,L). In this situation, mass from the layer below the top layer will be added to the top layer, and the new thickness TSED(1,L) will be TSED(1,L) + (TACT(L)-TSED(1,L)). Mass fractions (PER) of both layers are then recalculated.

```
ELSEIF (TSED(1,L) < TACT(L) .AND. TSED(1,L) + TSED(SLLN(L),L) > TACT(L) .AND.
TAU(L) > TAUCOR(SLLN(L),L)) THEN
  FORALL (K=1:NSCM) PER(K,1,L) = (PER(K,1,L) * TSED(1,L) + PER(K,SLLN(L),L) *
(TACT(L) - TSED(1,L))) / TACT(L)
  TSED(SLLN(L),L) = TSED(SLLN(L),L) - (TACT(L) - TSED(1,L))
  TSED(1,L) = TACT(L)
```

3. If there is net erosion but the layer immediately below the top layer does not have enough mass to contribute to the active layer, s_sedzlj.f90 will incorporate all of the material in the present 'erodible' layer to the active layer in this time step and reassign the 'erodible' layer. However, the active layer thickness might not be the same as the calculated active layer thickness.

```
ELSEIF (TSED(1,L) < TACT(L) .AND. TSED(1,L) + TSED(SLLN(L),L) <= TACT(L) .AND.
TAU(L) > TAUCOR(SLLN(L),L)) THEN
  FORALL (K=1:NSCM)
    PER(K,1,L) = (PER(K,1,L) * TSED(1,L) + PER(K,SLLN(L),L) * (TSED(SLLN(L),L))) /
(TSED(1,L) + TSED(SLLN(L),L))
    PER(K,SLLN(L),L) = 0.0
  ENDFORALL
  TSED(1,L) = TSED(1,L) + TSED(SLLN(L),L) !add available sediment to layer
  TSED(SLLN(L),L) = 0.0 !no more sediment available below this
  LAYER(SLLN(L),L) = 0 !layer has been eliminated in the logical variable
  SLLN(L) = SLLN(L) + 1 !set next layer lower
  IF (SLLN(L) + 1 > KB) SLLN(L) = KB !do not allow specification of the next
lower layer to be below the bottom sediment layer
  ENDIF
ENDIF
ENDIF
```

After the sorting process is completed, s_sedzlj.f90 calculates erosion. If a particular layer has eroded away or if the cell is in a state of deposition only, s_sedzlj.f90 identifies which sediment layers are still intact before reformulating bed properties.

```
ALL_LAYERS: DO LL=1,KB !go through all sediment layers so that they are
properly eroded and sorted
IF (LAYER(LL,L) == 0) CYCLE !if the layer is gone don't consider it
IF (LAYER(1,L) == 1 .AND. LL /= 1) EXIT !if it is depositional, there is no need to
consider erosion
IF (LL /= 1) THEN
  IF (LAYER(LL-1,L) == 1) EXIT
ENDIF
```

Before erosion calculations, the active (top) layer is re-characterized because the sediment in the active layer may have been adjusted. Average particle size (D50AVE; μm) of the layer is calculated using particle diameters (D50; μm) and weight fractions (PER). From D50AVE, critical shear stress (TAUCRIT) for the bed layer can be linearly interpolated from SEDflume

interpolant size class (SCLOC) and the corresponding critical shear for each interpolant (TAUCRITE). In the event that all the sediment bed layers are eroded away, the critical shear stress is set to 1.0×10^6 dynes/cm², so that there is no erosion of sediment into the water column.

```

D50AVG(L)=SUM(PER(1:NSCM,LL,L)*D50(1:NSCM))
DO K=1,NSICM-1
  IF(D50AVG(L)>=SCLOC(K).AND.D50AVG(L)<SCLOC(K+1))THEN
    NSCD(1)=SCLOC(K)
    NSCD(2)=SCLOC(K+1)
    NSC0=K
    NSC1=K+1
    EXIT
  ENDIF
ENDDO
IF(LL==1.OR.LL==2)THEN !for active layer
  TAUCRIT(L)=TAUCRITE(NSC0)+(TAUCRITE(NSC1)-TAUCRITE(NSC0))/(NSCD(2)
    -NSCD(1))*(D50AVG(L)-NSCD(1)) !interpolation
  TAUCOR(LL,L)=TAUCRIT(L)
ELSE
  SN01=TSED(LL,L)/TSED0(LL,L) !weighting factor 1 for interpolation
  SN11=(TSED0(LL,L)-TSED(LL,L))/TSED0(LL,L) !weighting factor 2
  IF(LL+1<=KB)THEN !Avoid array exceedance when LL=KB and used in
TAUCOR(LL+1,L)
  TAUCRIT(L)=SN01*TAUCOR(LL,L)+SN11*TAUCOR(LL+1,L) !
ELSE
  TAUCRIT(L)=1.0e6 !set arbitrarily high
ENDIF
ENDIF
ENDIF

```

The total erosion rate is interpolated across layer thicknesses and shear stresses. Linear interpolation is used to calculate the erosion rate at a specified shear stress (Jones 2001, eq. 3.3):

$$E(\tau) = \left(\frac{\tau_{i+1} - \tau}{\tau_{i+1} - \tau_m} \right) E_m + \left(\frac{\tau - \tau_i}{\tau_{i+1} - \tau_i} \right) E_{i+1}. \quad (24)$$

The subscripts i indicate data for a shear stress less than τ and $(i + 1)$ indicates a shear stress larger than τ . For example, consider a newly deposited sediment layer having a D50AVE of 150 μm . If the model calculates a shear stress (TAU) of 30.0 dynes/cm² (3.0 Pa), then the erosion rate (ERATEMOD) must be somewhere between the four values in Figure 3, bed.sdf (6.60E-5, 4.66E-4, 5.97E-5, and 5.97E-4 cm/s).

Because erosion rate varies rapidly with flow condition and bed properties, it is logarithmically interpolated from SEDflume data as a function of depth (Jones 2001, eq. 3.4):

$$\ln[E(T)] = \left(\frac{T_0 - T}{T_0} \right) \ln(E^{j+1}) + \frac{T}{T_0} \ln(E^j). \quad (25)$$

where T (m) is the actual layer thickness, T_0 (m) is the initial layer thickness, the superscripts j and $j+1$ denote data for the interface at the top and the bottom of the initial layer, respectively.

There are two kinds of erosion rate (ERATEMOD) calculations in this code: one for the initial bed erosion rate and another for erosion rate of newly deposited materials. The calculation for initial bed erosion rate, s_sedzlj.f90 first identifies the shear stress with respect to the erosion

rate data table from `erate.sdf` in Figure 1, by comparing the shear stress (TAU) with the shear stresses (TAULOC) at which erosion rate data (ERATE) are measured. When the closest value of TAULOC to TAU is found, `s_sedzlj.f90` begins calculating the modified erosion rate (ERATEMOD; cm/s) using the log-linear interpolation scheme described above. In cases where all of the sediment in the last layer is eroded away, ERATE is set to 10^{-9} cm/s and ERATEMOD = 0.

```

IF (TAU(L) < TAUCRIT(L)) EXIT
!Find the upper and lower limits of the Shear Stress for the interpolation
DO K=1, ITBM-1
  IF (TAU(L) >= TAULOC(K) .AND. TAU(L) < TAULOC(K+1)) THEN
    TAUDD(1) = TAULOC(K)
    TAUDD(2) = TAULOC(K+1)
    NTAU0 = K
    NTAU1 = K+1
    EXIT
  ENDIF
ENDDO
!Interpolate the erosion rates for shear stress and depth.
!This utilizes normal SEDflume data for deeper layers.
IF (LL > 2) THEN !calculate erosion rates of deeper layers
  SN00 = (TAUDD(2) - TAU(L)) / (TAUDD(2) - TAUDD(1)) !weighting factor 1 for
interpolation
  SN10 = (TAUDD(1) - TAU(L)) / (TAUDD(1) - TAUDD(2)) !weighting factor 2
  SN01 = TSED(LL, L) / TSED0(LL, L) !weighting factor 3
  SN11 = (TSED0(LL, L) - TSED(LL, L)) / TSED0(LL, L) !weighting factor 4
  IF (LL+1 <= KB) THEN !modeled erosion rate

ERATEMOD(L) = (SN00 * EXP(SN11 * LOG(ERATE(LL+1, L, NTAU0))) + SN01 * LOG(ERATE(LL, L, NTAU0
)))
&
+SN10 * EXP(SN11 * LOG(ERATE(LL+1, L, NTAU1))) + SN01 * LOG(ERATE(LL, L, NTAU1))) * BULKDENS
S(LL, L)
  ELSE !do not allow erosion through the bottom layer
    ERATEMOD(L) = (SN00 * EXP(SN11 * LOG(1e-9)) + SN01 * LOG(ERATE(LL, L, NTAU0)))
    & +SN10 * EXP(SN11 * LOG(1e-9)) + SN01 * LOG(ERATE(LL, L, NTAU1))) * BULKDENS(LL, L)
  ENDIF

```

For erosion of newly deposited sediments (layer 1 or 2), erosion rate (ENRATE) from `bed.sdf` in Figure 3 are interpolated to calculate the erosion rate (ERATEMOD).

```

ELSE
  NSCTOT = NSCD(2) - NSCD(1)
  D50TMPP = D50AVG(L) - NSCD(1)
  SN00 = (TAUDD(2) - TAU(L)) / (TAUDD(2) - TAUDD(1))
  SN10 = (TAUDD(1) - TAU(L)) / (TAUDD(1) - TAUDD(2))
  SN01 = D50TMPP / NSCTOT
  SN11 = (NSCTOT - D50TMPP) / NSCTOT
  ERATEMOD(L) = (SN00 * EXP(SN11 * LOG(ENRATE(NSC0, NTAU0)))
    + SN01 * LOG(ENRATE(NSC1, NTAU0)))
    + SN10 * EXP(SN11 * LOG(ENRATE(NSC0, NTAU1))) +
    + SN01 * LOG(ENRATE(NSC1, NTAU1))) * BULKDENS(LL, L)
ENDIF

```

Once the erosion rate (ERATEMOD) is interpolated based on the layer-averaged particle size (D50AVE) in the cell, the amount of material eroded per time step (EBD) is calculated by multiplying the erosion rate (ERATEMOD) by the time step (DT).

$$EBD(LL, L) = ERATEMOD(L) * DT$$

The following variables are calculated once the total sediment eroded per time step (EBD; g/cm²) is determined: total erosion at each cell for each size class (E) and total erosion per layer for each size class (ELAY). Top layer thickness (TTEMP) is also recalculated. If shear stress (TAU) is less than critical shear (TCRE) then all erosion values (E and ELAY) are set to zero and top layer thickness (TTEMP) is calculated without layer erosion (ELAY).

WHERE (TAU(L) >= TCRE(1:NSCM))

$$E(1:NSCM, L) = E(1:NSCM, L) + PER(1:NSCM, LL, L) * EBD(LL, L)$$

$$ELAY(1:NSCM) = PER(1:NSCM, LL, L) * EBD(LL, L)$$

$$TTEMP(1:NSCM, L) = PER(1:NSCM, LL, L) * TSED(LL, L) - ELAY(1:NSCM)$$

ELSEWHERE

$$E(1:NSCM, L) = 0.0$$

$$ELAY(1:NSCM) = 0.0$$

$$TTEMP(1:NSCM, L) = PER(1:NSCM, LL, L) * TSED(LL, L)$$

ENDWHERE

If amount of sediment eroded is greater than thickness of the bed layer (i.e., TTEMP < 0), thickness of the top layer (TTEMP) is set to zero and erosion at each cell for each size class (E) is set equal to the amount of sediment available in the eroding bed layer. Furthermore, sediment eroded from the layer (ELAY) is set to be the whole layer's thickness (PER*TSED).

WHERE (TTEMP(1:NSCM, L) < 0.0)

$$TTEMP(1:NSCM, L) = 0.0$$

$$E(1:NSCM, L) = E(1:NSCM, L) - PER(1:NSCM, LL, L) * EBD(LL, L)$$

$$+ PER(1:NSCM, LL, L) * TSED(LL, L)$$

$$ELAY(1:NSCM) = PER(1:NSCM, LL, L) * TSED(LL, L)$$

ENDWHERE

The total sediments eroded from the top (active) layer (ESED) and per cell (ETOTO) are calculated in units of g/cm².

$$ESED = SUM(ELAY(1:NSCM))$$

$$ETOTO(L) = SUM(E(1:NSCM, L))$$

Once erosion information is calculated, thickness of the top layer (TSED) and mass fractions (PER) is updated. If active layer thickness is less than 10⁻³ cm, TSED becomes zero and the top layer is eroded away (i.e., LAYER(LL, L) = 0). Otherwise, top layer thickness is set equal to calculated layer thickness after erosion (TEMP; cm), and the mass fraction (PER) is also updated accordingly.

$$TEMP = TSED(LL, L) - ESED$$

IF (TEMP <= 1.0E-3) **THEN**

$$TSED(LL, L) = 0.0$$

$$LAYER(LL, L) = 0$$

$$**FORALL** (K=1:NSCM) PER(K, LL, L) = 0.0$$

ELSE

$$TSED(LL, L) = TEMP$$

$$**FORALL** (K=1:NSCM) PER(K, LL, L) = TTEMP(K, L) / TSED(LL, L)$$

ENDIF

If change in morphology is considered (i.e., IMORPH = 0), then height of the bed (HBED; m) is calculated using the layer's thickness (TSED).

```
IF (IMORPH==0) HBED(L,LL)=0.01*TSED(LL,L)/BULKDENS(LL,L)
ENDDO ALL_LAYERS
```

Once the total erosion is calculated, sediment bed fluxes (QBSED and BED_SED_FLX; g/cm²) are divided into bedload (EBL) and suspended load (ESUS; g/cm²) using the probability for suspension (PSUS). When bedload transport is inactive (BLFLAG =0), only suspended sediment flux (BED_SED_FLX) is calculated.

```
EBL(L)=0.0
ESUS(L)=0.0
DO K=1,NSCM
  IF (BLFLAG(L,K)==1) THEN
    QBSED(L,K)=(1.0-PSUS(L,K))*E(K,L)-DBL(K,L)
    BED_SED_FLX(L,K)=PSUS(L,K)*E(K,L)-D(K,L)
  ELSE
    BED_SED_FLX(L,K)=E(K,L)-D(K,L)
  ENDF
ENDDO
EBL(L)=SUM((1.0-PSUS(L,1:NSCM))*E(1:NSCM,L),BLVEL(L,1:NSCM)>0.0)
ESUS(L)=SUM(PSUS(L,1:NSCM)*E(1:NSCM,L),BLVEL(L,1:NSCM)>0)
```

After calculating erosion information, s_sedzlj.f90 assigns EFDC variables. This is for backward compatibility with EFDC and the visualization program EFDC_Explorer. ETOTO is the total amount of eroded sediment. To convert this quantity into an erosion rate, s_sedzlj.f90 divides ETOTO by the time step (DT).

```
!Flux calculations for EFDC
!Convert BED_SED_FLX from g/cm^2 to g/m^2*s
WDTDZ=DT*HPI(L)*DZIC(1)
SEDF(L,0,1:NSCM)=BED_SED_FLX(L,1:NSCM)*10000.0/DT
SED(L,1,1:NSCM)=SEDS(L,1,1:NSCM)+(SEDF(L,0,1:NSCM)-SEDF(L,1,1:NSCM))*WDTDZ
ETOTO(L)=ETOTO(L)/(DT)
```

S MORPH.f90

Erosion and deposition may change the geometry sediment bed, which may in turn impact the hydrodynamics. Layer thickness (TSED), bulk density (BULKDENS), and previous bed height (HBEDA; m) are used to calculate change in bed height (DELBED; m).

BELV1 (m) and BELV (m) define the sediment-bed water column interface location for the previous and current time step, respectively. HP is the water column height, P (m^3/s) is the product of water surface elevation and acceleration due to gravity, and HBEDA is the total bed height. The sediment-bed water column interface location (BELV) and the water column depth (HP) vary according to the change in bed height (DELBED).

```
DO L=2, LA
  DELBED(L)=0.01*SUM(TSED(1:KB,L)/BULKDENS(1:KB,L))-HBEDA(L)
  BELV1(L)=BELV(L)
  HTMP(L)=HP(L)
  H1P(L)=HP(L)
  P1(L)=P(L)
  HBEDA(L)=0.01*SUM(TSED(1:KB,L)/BULKDENS(1:KB,L))
  HBED(L,1:KB)=0.01*TSED(1:KB,L)/BULKDENS(1:KB,L)
  BELV(L)=ZELBEDA(L)+HBEDA(L)
  HP(L)=HP(L)+DELBED(L)
```

ENDDO

The inverse of the water depth (HPI; m^{-1}) and the water flux (QMORPH; $\text{kg}/\text{m}^2/\text{s}$) in/out of the water column are also calculated using the current column height (HP) and previous water column heights (H1P; m).

```
DO L=2, LA
  HPI(L)=1.0/HP(L)
  QMORPH(L)=DELTI*DXYP(L)*(HP(L)-H1P(L))
```

ENDDO

s_morph.f90 also writes an output file for drying cells and error checking.

```
ITMP=0
DO L=2, LA
  IF(HP(L)<0.0) THEN
    IF(ABS(H1P(L))>=HWET) THEN
      ITMP=1
      WRITE(8, "('NEG DEPTH DUE TO MORPH CHANGE', 2I5, 12F12.5)") IL(L), JL(L),
        HBED1(L, KBT(L)), HBED(L, KBT(L)), BELV1(L), BELV(L), DELT,
        QSBDBTOP(L), QWBDBTOP(L), HBEDA(L)
      WRITE(8, "('NEG DEPTH DUE TO MORPH CHANGE',
2I5, 12F12.5)") L, KBT(L), (HBED(L, K), K=1, KBT(L))
    ELSE
      HP(L)=0.9*HDRV
    ENDIF
  ENDIF
ENDDO
IF(ITMP==1) THEN
  CALL RESTOUT(1)
  IF(NDRYSTP<0) THEN
    OPEN(1, FILE='DRYLOSS.OUT')
    CLOSE(1, STATUS='DELETE')
    OPEN(1, FILE='DRYLOSS.OUT')
```

```

DO L=2, LA
  IF (VDWASTE(L) > 0.0) THEN
    TMPVAL = VDWASTE(L) / DXYP(L)
    WRITE(1, '(2I6, 4E14.6)') IL(L), JL(L), VDWASTE(L), TMPVAL, QDWASTE(L)
  ENDIF
ENDDO
CLOSE(1)
ENDIF
STOP
ENDIF

```

Changes in morphology cause change in water column depth. Hence, the concentrations of all transported materials must be adjusted.

```

IF (ISTRAN(1) > 0) FORALL (K=1:KC, L=2:LA) SAL(L, K) = HTMP(L) * SAL(L, K) / HP(L)
IF (ISTRAN(2) > 0) FORALL (K=1:KC, L=2:LA) TEM(L, K) = HTMP(L) * TEM(L, K) / HP(L)
IF (ISTRAN(3) > 0) FORALL (K=1:KC, L=2:LA) DYE(L, K) = HTMP(L) * DYE(L, K) / HP(L)
IF (ISTRAN(4) > 0) FORALL (K=1:KC, L=2:LA) SFL(L, K) = HTMP(L) * SFL(L, K) / HP(L)
IF (ISTRAN(5) > 0) FORALL (NT=1:NTOX, K=1:KC, L=2:LA) TOX(L, K, NT) = HTMP(L) * TOX(L, K, NT)
                                                                /HP(L)
IF (ISTRAN(6) > 0) FORALL (NS=1:NSCM, K=1:KC, L=2:LA) SED(L, K, NS) = HTMP(L) * SED(L, K, NS)
                                                                /HP(L)
IF (ISTRAN(7) > 0) FORALL (NS=1:NSND, K=1:KC, L=2:LA) SND(L, K, NS) = HTMP(L) * SND(L, K, NS)
                                                                /HP(L)

```

Variable Dictionary

The table below lists the SNL-EFDC variables in alphabetical order with a brief description. Variables from EFDC are denoted as such in the description.

Variable name	Description
BEDLOAD	Flag for bedload transport (1 = yes)
BDEN	Bulk density for each sediment layer (g/cm ³)
BED_SED_FLX	Suspended sediment flux (g/cm ²)
BEDLINIT	Initial bed thickness (m)
BELV	Sediment-bed water column interface location (m)
BLVEL	Bedload velocity (cm/s)
BULKDENS	Mass of solids per unit volume (g/cm ³)
CBL	Bedload concentration (g/cm ³)
CSEDVR	Van Rijn's equilibrium bedload concentration (g/cm ³)
CSEDSS	Sediment eroded into bedload (g/cm ³)
CTB	Concentration at sediment bed (g/cm ³)
D	Deposition from suspended load (g/cm ²)
D50	Particle diameter for each size class (μm)
D50AVE	Average particle size (μm)
DBL	Deposition from bedload (g/cm ²)
DELBED	Change in the sediment bed height, EFDC (m)
DELTI	Inverse time step (s ⁻¹)
DEP	Total deposition of each cell (g/cm ²)
DISTAR	Non-dimensional particle diameter for each size class
DT	Time step (s)

Variable name	Description
DWS	Settling speed
DZBL	Bedload height (cm)
DZC	Water-layer thickness factor
E	Total erosion of each cell per time step (g/cm^2)
EORATE	Initial erosion rate for each layer subject to TAULOC (cm/s)
EBD	Total sediment eroded per time step (g/cm^2)
EBL	Total erosion into bedload per time step (g/cm^2)
ELAY	Total erosion of the top layer per time step (g/cm^2)
ENRATE	Erosion rate for newly deposited bed (cm/s)
ERATE	EORATE (cm/s)
ERATEMOD	Erosion rate for each layer ($\text{g}/\text{cm}^2\text{s}$)
ESED	Total erosion from each layer per time step (g/cm^2)
ESUS	Total erosion into suspended load per time step (g/cm^2)
ETOTO	Total erosion of all size classes (g/cm^2)
HBED	Bed layer thickness, EFDC (m)
HP	Water column depth (m)
HT	Water column depth (cm)
IFWAVE	Flag that activates wind-driven surface waves
IMORPH	Flag that activates bed morphology calculations (1 = yes)
ISEDTIME	Number of time steps before sediment transport is activated
ISMORPH	Flag that activates bed morphology calculations (1 = yes)
ITBM	Number of different SEDflume shear-stress interpolants to calculate erosion of newly deposited sediments
KB	Maximum number of sediment layers

Variable name	Description
MAXDEPLIMIT	Maximum fraction of sediment that can deposit per time step
NCALC_BL	Flag that activates bedload sediment transport (1 = yes)
NEQUIL	Flag to activate adsorption of toxics onto sediments (1 = yes)
NSEDFLUME	Flag to activate SEDZLJ routines (1 or 2 = yes), set by the EFDC variable IWRSP = 98 or 99
NSCM	Number of sediment size classes, from EFDC
NSICM	Number of size classes of interpolants to estimate erosion rates of newly deposited sediments
P	PNEW
PER	PNEW/100.0
PNEW	Mass percentage of each size class in layer
PROB	Probability of suspended load deposition
PROBVR	Probability of deposition from bedload
PSUS	Percentage of total erosion into suspended load
QBSSED	Net flux into bedload per time step (g/cm^2)
QSBDBTOP	Total volumetric sediment flux, EFDC ($\text{kg}/\text{m}^2\text{s}$)
RHO	Density of water, EFDC (kg/m^3)
SCLOC	Particle size for new deposited bed erosion data (μm)
SED	Suspended sediment concentration (kg/m^3)
SEDF	Sediment flux ($\text{kg}/\text{m}^2\text{s}$)
SEDS	SED, used for vertical turbulent dispersion and settling calculations (kg/m^3)
SEDDENS	Sediment density (g/cm^3)
SMASS	Sediment mass available for deposition (g/cm^2)
SSGI	Inverse specific gravity

Variable name	Description
SURFACE	Sediment layer that corresponds to the active layer
TACT	Active layer thickness (g/cm^2)
TACTM	Active layer thickness multiplier
TAU	Total shear stress (dynes/cm^2)
TAUB	Total shear stress, EFDC (Pa)
TAUCONST	Parameter that, when greater than zero, sets the shear stress to a constant value in the flow (dynes/cm^2)
TAUCOR	Critical shear stress of the initial sediment bed (dynes/cm^2)
TAUCR	Critical shear stress for erosion for each size class. (dynes/cm^2)
TAUCRIT	Critical shear stress for erosion (dynes/cm^2)
TAUCRITE	Critical shear stress for erosion for new deposited bed (dynes/cm^2)
TAULOC	Applied shear stress from erosion data (dynes/cm^2)
TAUTEMP	Critical shear stress for the erosion of each layer (dynes/cm^2)
TCRDEP	Critical shear stress for suspension for each size class. (dynes/cm^2)
TCRE	Critical shear for erosion (dynes/cm^2)
TCRSUS	Critical shear for suspension (dynes/cm^2)
TEMP	Calculated layer thickness after erosion (cm)
TRANS	Dimensionless transport parameter for bedload calculations (van Rijn).
TSED	Current thickness of each layer (g/cm^2)
TSED0	TSED0S (g/cm^2)
TSED0S	Initial thickness of each sediment layer (g/cm^2)
TTEMP	Temporary thickness of a given layer (g/cm^2)
U	x -component of the hydrodynamic velocity (m/s)

Variable name	Description
UBL	x -component of the bedload velocity (cm/s)
USTAR	Shear velocity (m/s)
USW	Ratio of shear velocity to settling velocity
V	y -component of the hydrodynamic velocity (m/s)
VAR_BED	Flag that activates the variable sediment bed option (1 = yes)
VBL	y -component of the bedload velocity (cm/s)
VDRBED	Initial void ratio in each bed layer
VDRBEDO	Sediment ratio in each bed layer
VELMAG	Velocity magnitude (m/s)
VZDIF	Vertical diffusion for suspended sediments (cm ² /s)
WATERDENS	Density of water (g/cm ³)
WSETA	Sediment settling velocity, EFDC (m/s)
WVEL	Sediment vertical velocity, EFDC (m/s)
XBLFLUX	x -component of the horizontal bedload flux per time step (g/cm ³)
YBLFLUX	y -component of the horizontal bedload flux per time step (g/cm ³)
ZBSKIN	Roughness parameter (μ m)

Special Notes

Currently, `s_sedzlj.f90` does not incorporate bedload sediment flux through boundaries of the domain. This feature will soon be added to SNL-EFDC. Bed consolidation is also an expected upgrade.

There are a few issues in EFDC not mentioned in the user's manual that we document in this section to assist users of SNL-EFDC.

The EFDC routine `sser.inp` divides the salinity value by the mass of each water layer. This is somewhat confusing because salinity is an intrinsic variable.

Also, `QSSER` from `efdc.inp` is added to `QSER.inp` and it is easy to neglect the fact that both flow inputs may be active in a model.

References

- Abramowitz, M., and Stegun, I. (1972). *Handbook of Mathematical Functions*, Washington, D.C.
- Cheng, N. S. (1997). "Simplified settling velocity formula for sediment particle." *Journal of Hydraulic Engineering-ASCE*, 123(2), 149-152.
- Christoffersen, J., and Jonsson, I. (1985). "Bed friction and dissipation in a combined current and wave motion." *Ocean Engineering*, 17(4), 479-494.
- Fischer, H., List, E., Koh, R., Imberger, J., and Brooks, N. (1979). *Mixing in Inland and Coastal Waters*, Academic Press, New York, NY.
- Gessler, J. (1967). "The beginning of bedload movement of mixtures investigated as natural armoring in channels."
- Guy, H. P., Simons, D. B., and Richardson, E. V. (1966). "Summary of alluvial channel data from flume experiments." *Geological Survey Professional Paper*, 462(I).
- Hamrick, J. M. (1992). "A Three-Dimensional Environmental Fluid Dynamics Computer Code: Theoretical and Computational Aspects." The College of William and Mary.
- Hamrick, J. M. (1996). "User's Manual for the Environmental Fluid Dynamics Computer Code." Virginia Institute of Marine Sciences, Gloucester Point, Virginia.
- Jones, C. (2001). "SEDZLJ: A Sediment Transport Model," University of California, Santa Barbara.
- Jones, C., and Lick, W. (Year). "Sediment erosion rates: Their measurement and use in modeling." *The Texas A&M Dredging Seminar*, College Station, Texas, 1-15.
- Krone, R. B. (1962). "Technical Report DA-04-203 CIVENG-59-2." US Army Corps of Engineers, San Francisco, CA.
- Roberts, J. D., Jepsen, R. A., and James, S. C. (2003). "Measurements of sediment erosion and transport with the Adjustable Shear Stress Erosion and Transport Flume." *Journal of Hydraulic Engineering-Asce*, 129(11), 862-871.
- Roberts, J. D., Jepsen, R. A., and Lick, W. (1998). "Effects of particle size and bulk density on erosion of quartz particles." *Journal of Hydraulic Engineering-ASCE*, 124(12), 1261-1267.
- Soulsby, R. L. (1997). *Dynamics of Marine Sands*, Thomas Telford, London, UK.
- Van Niekerk, A., Vogel, K., Slingerland, R., and Bridge, J. (1992). "Routing of heterogeneous sediments over movable bed: model development." *J. Hydr. Engr.*, 118(2), 246-263.
- van Rijn, L. C. (1984a). "Sediment transport, Part I: Bed load transport." *Journal of Hydraulic Engineering-ASCE*, 110(11), 1613-1641.
- van Rijn, L. C. (1984b). "Sediment transport, Part II: Suspended load transport." *Journal of Hydraulic Engineering-ASCE*, 110(11), 1613-1641.