# *EFDC Technical Memorandum*

# *Theoretical and Computational Aspects of the Generalized Vertical Coordinate Option in the EFDC Model*

***Prepared for:***

***US Environmental Protection Agency, Region 4***
***Atlanta, GA***

***Prepared by:***

***Tetra Tech, Inc.***
***10306 Eaton Place***
***Suite 340***
***Fairfax, Virginia 22030***

***March 2006***

# Table of Contents

## Acknowledgement

The development of the generalized vertical coordinate option in EFDC was supported by the U.S. Environmental Protection Agency, Region 4. Mr. James Greenfield was the US EPA work assignment manager and Mr. Steven Davie was the Tetra Tech work assignment manager. Dr. John M. Hamrick of Tetra Tech, implemented the EFDC code modifications and is the author of this report. Drs. Sen Bai, Silong Lu, Hugo Rodriguez, and Rui Zou, all of Tetra Tech, provided valuable assistance in testing the EFDC code implementation.

# 1. Introduction

This report summarizes theoretical and computational aspects of the generalized vertical coordinate option in the Environmental Fluid Dynamics Computer Code (EFDC). It supplements the theoretical and computational description of the basic EFDC hydrodynamic and transport model components. The EFDC model was originally formulated with a sigma or stretched vertical coordinate. In the sigma coordinate formulation, the number of vertical layers is the same at all horizontal locations in the model grid. Although this formulation is widely accepted, conceptually attractive and adequate for a large range of applications, there are a number of application classes where a traditional Z or physical vertical grid is desirable, such as deep reservoirs which rapid and large lateral bathymetric changes. There are also applications where the ability to use a combination of sigma and physical Z vertical layering in different regions of the horizontal domain would be desirable. An example would be a deep navigation channel in an otherwise shallow estuary. The sigma grid vertical grid formulation may also be subject to internal pressure gradient errors (Mellor *et al*., 1994, 1998) providing another motivation for having the options to the sigma formulation.

There are a number of alternatives for mixing both sigma and physical vertical coordinate systems in a single domain. These include the sigma over Z grid (Martin, 2000) and the generalized sigma grid, often referred to as the S grid (Ezer and Mellor, 2004). A somewhat different approach has been taken here to arrive at hybrid or generalized vertical grid. The approach allows the horizontal model domain to be partitioned into sigma regions and what can be referred to as laterally constrained, localized sigma regions (LCL sigma). In the LCL region, the number of active vertical layer is variable, while in the sigma region, the number of vertical layers is constant. The vertical coordinate transformation in the LCL sigma region is based on a minor modification of the rescaled height coordinate transformation (Stacey *et al*., 1995; Adcroft and Campin, 2004). Although the LCL transformation includes the sigma transformation as a special case, the vertical grid behavior has strong similarities with the traditional Z vertical grid, with the advantage of the free surface being a constant coordinate surface.

This report is organized as follows. Chapter 2 summarizes the generalized vertical coordinate transform leading to a hybrid of the sigma and LLC sigma coordinates now available in EFDC. Chapter 3 summarizes the generalized forms of the momentum, continuity, and transport equations. Chapter 4 describes the configuration of the generalized vertical coordinate option in the EFDC including input file templates and options for plotting vertical plane grid sections and output variable fields. Cited references are listed in Chapter 5. Figures are grouped in Chapter 5.

# 2. Generalize Vertical Coordinate Transformations

This chapter presents a derivation of the generalized vertical coordinate transformation implemented in the EFDC. The transform subsequently referred to as a laterally constrained localized (LCL) sigma transform is general in the sense that it includes the standard sigma transformation as a special case. The implementation is hybrid in the sense that LCL sigma and standard sigma vertical grids can be mixed on a sub-domain basis in a single model configuration.

## 2. 1 Derivation of the Transformation

The most general free surface following coordinate transformation can be written as

$$Z = Z_s + S\left(X, Y, T, z\right) \tag{1}$$

where

$$\begin{aligned}
&Z : Cartesian\ Physical\ Vertical\ Coordinate \\
&Z_s\left(\mathbf{X}, T\right) : Cartesian\ Coordinate\ of\ \ Free\ Surface \\
&z : Dimensionless\ General\ Vertical\ Coordinate \\
&\mathbf{X} = \left(X, Y\right) : Horizontal\ \ Coordinates\left(Before\ Transformation\right) \\
&T : Time\left(Before\ Transformation\right) \\
&S : Continuous\ and\ /\ or\ \ Discrete\ Transform\ Function
\end{aligned} \tag{2}$$

The function $S$ must satisfy the condition that transformed vertical coordinate, $z$ is constant when $Z$ is equal to $Z_s$, (Mellor *et al*., 2004). The laterally constrained localized sigma vertical coordinate transformation is given by

$$Z = Z_s - \frac{\left(Z_{sref} - Z_{bmin}\right)}{\left(Z_{sref} - Z_b\right)}\left(Z_s - Z_b\right)\left(1 - z\right) \tag{3}$$

where

$$\begin{aligned}
&Z_{sref} : Reference\ Cartesian\ Coordinate\ of\ \ Free\ Surface \\
&Z_b\left(\mathbf{X}\right) : Cartesian\ Coordinate\ of\ \ Bed \\
&Z_{bmin} : Minimum\ Cartesian\ Coordinate\ of\ \ Bed \\
&z : Dimensionless\ General\ Vertical\ Coordinate\ (0,1)
\end{aligned} \tag{4}$$

The transformation (3) is based on a slight modification of the rescaled height coordinate transformation (Stacey *et al*., 1995; Adcroft and Campin, 2004). It is readily verified that

(3) satisfies the condition that z is equal to 1 at the free surface were $Z$ equals $Z_s$. Also note that at horizontal locations were $Z_b$ is equal to $Z_{bmin}$, the LCL coordinate of the bed, $z_b$, is zero. At locations were $Z_b$ is greater than $Z_{bmin}$, the LCL coordinate of the bed will be greater than zero. As will subsequently be shown, $Z_{sref}$ in conjunction with $Z_{bmin}$ serve to initialize the vertical layer distribution.

The LCL transformation (3) can be written as

$$Z = Z_s - \lambda\left(Z_s - Z_b\right)\left(1 - z\right) = Z_s - \lambda H\left(1 - z\right) \tag{5}$$

that has the inverse transformation

$$z = \frac{\lambda\left(Z_s - Z_b\right) + Z - Z_s}{\lambda\left(Z_s - Z_b\right)} = \frac{\lambda H + Z - Z_s}{\lambda H} \tag{6}$$

where

$$H = Z_s - Z_b \tag{7}$$

is the depth of the water column and

$$\lambda = \frac{Z_{sref} - Z_{bmin}}{Z_{sref} - Z_b}\,;\ LCL\ Sigma \tag{8}$$

for the LCL sigma coordinate transformation. For

$$\lambda = 1\,;\ Standard\ Sigma \tag{9}$$

the transformations (5 and 6) revert to the 0 to 1 form of the standard sigma transformation.

The parameter $\lambda$ can be alternately be defined by differentiating (5)

$$dZ = \lambda H dz \tag{10}$$

and integrating over the water column depth, noting that in the LCL coordinate system the dimensional vertical coordinate can have a non-zero value at the bed.

$$\int_{Z_b}^{Z_s} dZ = \lambda H \int_{z_b}^{1} dz \tag{11}$$

The integral (11) gives

$$\lambda = \frac{1}{1 - z_b} \tag{12}$$

allowing $\lambda$ to be alternately defined without use of $Z_{sref}$ and $Z_{bmin}$.

## 2. 2 Initialization of the Vertical Layering

The vertical layering in the LCL sigma coordinate can be initialized by two approaches. The first approach involves defining a sequence of laterally constraining horizontally invariant elevations in the physical *Z* coordinate. This approach can optionally be used to define rounded local bed elevations in the horizontal model grid when initial bed elevations are defined by any of a number of bathymetry interpolation options in the EFDC grid generation programs, GEFDC and VOGG.

For a maximum of *KC* vertical layers, *KC*+1 constraining elevations are required with the lowest (*k*=0) and the highest (*k*=*KC*) required to be

$$\begin{aligned} Z_0^{LC} &= Z_{bmin} \\ Z_{KC}^{LC} &= Z_{sref} \end{aligned} \tag{13}$$

where the index *k* goes from 0 to *KC*. For a constant increment

$$\Delta_Z^{LC} = \frac{Z_{sref} - Z_{bmin}}{KC} \tag{14}$$

between the constraining elevations, the rounded physical bottom elevation, $Z_{br}$ at a horizontal location having *KL* active vertical layers is given by

$$Z_{br} = Z_{sref} - KL\Delta_Z^{LC} = Z_{sref} - KL\left(\frac{Z_{sref} - Z_{bmin}}{KC}\right) \tag{15}$$

Starting from a non-rounded set of bottom elevations, the local number of layers can be determined by locally selecting the *KL* whose rounded bottom elevation is closest to the non-rounded bottom elevation. Optionally, the initial non-rounded bottom elevations can be replaced with the rounded elevations. Using (8) is readily shown that

$$\lambda = \frac{KC}{KL} \tag{16}$$

for constant increments of the lateral constraining elevation. The LCL sigma coordinate of the bed, $z_b$, is when determined from (12). For the constant increment case it is also readily shown that the dimensionless LCL layer thicknesses are uniform and given by

$$\Delta_z = \frac{1}{KC} \tag{17}$$

The integral of (11) over the local depth of the water column

$$\int_{Z_b}^{Z_s} dZ = \int_{z_b}^{1} \lambda H dz = \lambda H \sum_{k=KC-KL+1}^{KC} \Delta_z = \lambda H \left( \frac{KL}{KC} \right) = H \tag{18}$$

defines the role of the scale factor in accounting for the number of local number of LCL layers in satisfying the vertical integral constraint of the coordinate transformation. The constant increment lateral constraining elevation initialization approach is illustrated by considering a simple example, with $Z_{bmin}$ equal to 0 and $Z_{sref}$ equal to 12 and a four- layer grid with the relevant results shown in Table 1.

For the more general case of unequal increments in the lateral constraining elevations, the physical increments are specified. Table 2 shows the same simple example with unequal lateral constraining elevations specified and the resulting LCL local scale factors, bottom elevations and layer thicknesses. For calculation of the discrete transformation in Table 2, the scale factor is determined using (8) and the LCL local bottom coordinate by (11), with the LCL layer thickness corresponding to the difference in LCL bottom layer increments.

The second approach to initialization involves specification of the LCL layer thicknesses, $\Delta_{z,k}$. The specification must be consistent with the maximum number of layers, *KC*, and the requirement that the sum of the LCL layer thicknesses equal one.

$$\sum_{k=1}^{KC} \Delta_{z,k} = 1 \tag{19}$$

The LCL bottom coordinates, corresponding to *KL* equal 1 to *KC*, local layers, are determined by

$$z_b(KL) = 1 - \sum_{k=KC-KL+1}^{KC} \Delta_{z,k} \tag{20}$$

The (*KL*) notation is used to associate values with locations have *KL* layers. The scale factor is determined using (12), which can be written as

$$\lambda(KL) = \frac{1}{1 - z_b(KL)} = \frac{1}{\sum_{k=KC-KL+1}^{KC} \Delta_{z,k}} \tag{21}$$

The corresponding rounded local bottom elevation follows from (8) and is

$$Z_{br}(KL) = Z_{sref} - \frac{Z_{sref} - Z_{bmin}}{\lambda(KL)} \tag{22}$$

Starting from a non-rounded set of bottom elevations, the local number of layers can be determined by locally selecting the *KL* whose rounded bottom elevation is closest to the non-rounded bottom elevation. Optionally, the initial non-rounded bottom elevations can be replaced with the rounded elevations. Table 3 shows the same simple example, but in this case the calculation proceeds from right to left start with an initial set of LCL layer thicknesses. Note also that the cases in Tables 1 and 2 can be interpreted in a right to left manner.

Table 1: Example of Vertical Layering for Constant Increments of the Lateral Constraint Elevations. The Constant Increment is the Difference Between Columns 3 and 2.

| Layer | Bottom LC Elev | Top LC Elev | Local Bottom Elev | Local Number of Layers | Scale Factor | Local $z_b$ for Local Bottom Elevation | LCL Layer Thick. |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 3 | 0 | 4 | 1 | 0 | 0.25 |
| 2 | 3 | 6 | 3 | 3 | 1.333 | 0.25 | 0.25 |
| 3 | 6 | 7.5 | 6 | 2 | 2 | 0.5 | 0.25 |
| 4 | 9 | 12 | 9 | 1 | 4 | 0.75 | 0.25 |

Table 2: Example of Vertical Layering for Variable Increments of the Lateral Constraint Elevations. The Variable Increments are the Differences Between Columns 3 and 2.

| Layer | Bottom LC Elev | Top LC Elev | Local Bottom Elev | Local Number of Layers | Scale Factor | Local $z_b$ for Local Bottom Elevation | LCL Layer Thick. |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 4.8 | 0 | 4 | 1 | 0 | 0.4 |
| 2 | 4.8 | 8.4 | 4.8 | 3 | 1.667 | 0.4 | 0.3 |
| 3 | 8.4 | 10.8 | 8.4 | 2 | 3.333 | 0.7 | 0.2 |
| 4 | 10.8 | 12 | 10.8 | 1 | 10 | 0.9 | 0.1 |

Table 3: Example of Vertical Layering for Variable Increments of the Lateral Constraint Elevations. The Constant Increment is the Difference Between Column 3 and 2.

| Layer | Bottom LC Elev | Top LC Elev | Local Bottom Elev | Local Number of Layers | Scale Factor | Local $z_b$ for Local Bottom Elevation | LCL Layer Thick. |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 4.2 | 0 | 4 | 1 | 0 | 0.35 |
| 2 | 4.2 | 7.2 | 4.2 | 3 | 1.5385 | 0.35 | 0.25 |
| 3 | 7.2 | 9.6 | 7.2 | 2 | 2.5 | 0.6 | 0.2 |
| 4 | 9.6 | 12 | 9.6 | 1 | 5 | 0.8 | 0.2 |

## 2. 3 Derivative Transformation Relationships

Implementation of the generalized LCL vertical coordinate system (x,y,z,t) requires derivative transformation relationships form the base (X,Y,Z,T) system. Since the existing horizontal coordinate system in the EFDC model is curvilinear and orthogonal, the base horizontal coordinates (X,Y) will be presumed to be curvilinear and orthogonal. Formally, the derivative transformation began with

$$\begin{aligned} X &= X\left(x, y, z, t\right) \\ Y &= Y\left(x, y, z, t\right) \\ Z &= Z\left(x, y, z, t\right) \\ T &= t\left(x, y, z, t\right) \end{aligned} \tag{23}$$

The derivatives in the (*X,Y,Z,T*) system can be written as

$$\begin{aligned} \frac{\partial}{\partial X} &= \frac{\partial}{\partial x} - \frac{\partial_x Z}{\partial_z Z}\frac{\partial}{\partial z} \\ \frac{\partial}{\partial Y} &= \frac{\partial}{\partial y} - \frac{\partial_y Z}{\partial_z Z}\frac{\partial}{\partial z} \\ \frac{\partial}{\partial Z} &= \frac{1}{\partial_z Z}\frac{\partial}{\partial z} \\ \frac{\partial}{\partial T} &= \frac{\partial}{\partial t} - \frac{\partial_t Z}{\partial_z Z}\frac{\partial}{\partial z} \end{aligned} \tag{24}$$

following Ezer and Mellor (2004). With *Z* defined by (5),

$$Z = Z_s - \lambda H\left(1 - z\right) \tag{5}$$

the partial Z derivatives are

$$\begin{aligned} \partial_x Z &= \partial_x Z_s - \left(1 - z\right)\partial_x\left(\lambda H\right) \\ \partial_y Z &= \partial_y Z_s - \left(1 - z\right)\partial_y\left(\lambda H\right) \\ \partial_z Z &= \lambda H \\ \partial_t Z &= \partial_t Z_s - \left(1 - z\right)\partial_t\left(\lambda H\right) \end{aligned} \tag{25}$$

with (24) becoming

$$
\begin{aligned}
\lambda H \frac{\partial \phi}{\partial X} &= \lambda H \frac{\partial \phi}{\partial x} - \left( \partial_x Z_s - (1-z) \partial_x (\lambda H) \right) \frac{\partial \phi}{\partial z} \\
\lambda H \frac{\partial \phi}{\partial Y} &= \lambda H \frac{\partial \phi}{\partial y} - \left( \partial_y Z_s - (1-z) \partial_y (\lambda H) \right) \frac{\partial \phi}{\partial z} \\
\lambda H \frac{\partial \phi}{\partial Z} &= \frac{\partial \phi}{\partial z} \\
\lambda H \frac{\partial \phi}{\partial T} &= \lambda H \frac{\partial \phi}{\partial t} - \left( \partial_t Z_s - (1-z) \partial_t (\lambda H) \right) \frac{\partial \phi}{\partial z}
\end{aligned}
\tag{26}
$$

for a generic dependent variable. Equations (26) can be written in the intermediate conservative forms

$$
\begin{aligned}
\lambda H \frac{\partial \phi}{\partial X} &= \partial_x (\lambda H \phi) - \phi \partial_x (\lambda H) - \left( \partial_x Z_s - (1-z) \partial_x (\lambda H) \right) \partial_z \phi \\
\lambda H \frac{\partial \phi}{\partial Y} &= \partial_y (\lambda H \phi) - \phi \partial_y (\lambda H) - \left( \partial_y Z_s - (1-z) \partial_y (\lambda H) \right) \partial_z \phi \\
\lambda H \frac{\partial \phi}{\partial Z} &= \partial_z \phi \\
\lambda H \frac{\partial \phi}{\partial T} &= \partial_t (\lambda H \phi) - \phi \partial_t (\lambda H) - \left( \partial_t Z_s - (1-z) \partial_t (\lambda H) \right) \partial_z \phi
\end{aligned}
\tag{27}
$$

and the final conservative forms

$$
\begin{aligned}
\lambda H \frac{\partial \phi}{\partial X} &= \partial_x (\lambda H \phi) - \partial_z \left( \left( \partial_x Z_s - (1-z) \partial_x (\lambda H) \right) \phi \right) \\
\lambda H \frac{\partial \phi}{\partial Y} &= \partial_y (\lambda H \phi) - \partial_z \left( \left( \partial_y Z_s - (1-z) \partial_y (\lambda H) \right) \phi \right) \\
\lambda H \frac{\partial \phi}{\partial Z} &= \partial_z \phi \\
\lambda H \frac{\partial \phi}{\partial T} &= \partial_t (\lambda H \phi) - \partial_z \left( \left( \partial_t Z_s - (1-z) \partial_t (\lambda H) \right) \phi \right)
\end{aligned}
\tag{28}
$$

Now consider the conservative form of the total or advective derivative in the base coordinate system

$$
\frac{\partial \left( m_x m_y \phi \right)}{\partial T} + \frac{\partial \left( m_y u \phi \right)}{\partial X} + \frac{\partial \left( m_x v \phi \right)}{\partial Y} + \frac{\partial \left( m_x m_y W \phi \right)}{\partial Z} = m_x m_y r_\phi \tag{29}
$$

where $u$ and $v$ are the horizontal velocities and $m_x$ and $m_y$ are the scale factors in the curvilinear, orthogonal horizontal coordinates which remain unchanged by the transformation (5) and $r$ is a source or sink term. The continuity equation consistent with (29) is

$$\frac{\partial\left(m_y u\right)}{\partial X}+\frac{\partial\left(m_x v\right)}{\partial Y}+\frac{\partial\left(m_x m_y W\right)}{\partial Z}=0 \tag{30}$$

that follows from setting $\phi$ to unity and the source term to zero. Multiplying (29) by $\lambda H$ gives after a lot of algebraic manipulation

$$\begin{aligned}&\partial_t\left(m_x m_y \lambda H \phi\right)+\partial_x\left(m_y \lambda H u \phi\right)+\partial_y\left(m_x \lambda H v \phi\right)\\&+\partial_z\left(m_x m_y w \phi\right)=m_x m_y \lambda H r_\phi\end{aligned} \tag{31}$$

where the vertical velocity, $w$, in the transformed coordinate system is

$$\begin{aligned}w=&W-\partial_t Z_s-(1-z)\partial_t(\lambda H)\\&-\frac{u}{m_x}\left(\partial_x Z_s-(1-z)\partial_x(\lambda H)\right)\\&-\frac{v}{m_y}\left(\partial_y Z_s-(1-z)\partial_y(\lambda H)\right)\end{aligned} \tag{32}$$

The continuity equation (30) becomes

$$\partial_t\left(m_x m_y \lambda H\right)+\partial_x\left(m_y \lambda H u\right)+\partial_y\left(m_x \lambda H v\right)+\partial_z\left(m_x m_y w\right)=0 \tag{33}$$

# 3. Transformed Model Equations

This section summarizes the transformed hydrodynamic and transport equations in the LCL coordinate system.

## 3. 1 Continuity Equations

The continuity equation (33) is generalized to include a source/sink term

$$\partial_t\left(m_x m_y \lambda H\right)+\partial_x\left(m_y \lambda H u\right)+\partial_y\left(m_x \lambda H v\right)+\partial_z\left(m_x m_y w\right)=Q \tag{34}$$

Integration of (34) over LCL coordinate layer *k* gives

$$\begin{aligned}\partial_t\left(m_x m_y \lambda H \Delta_k^c\right)+\partial_x\left(m_y \lambda H \Delta_k^c u_k\right)+\partial_y\left(m_x \lambda H \Delta_k^c v_k\right)\\ +\left(m_x m_y w\right)_{z=z_k^g}-\left(m_x m_y w\right)_{z=z_{k-1}^g}=Q_k\end{aligned} \tag{35}$$

where $\Delta_k^c$ is the dimensionless layer thickness, $z_k^g$ is the LCL sigma coordinate at the top of the layer, and $Q_k$ is the volume source or sink in layer *k*. Summing (35) over the local number of layers using

$$\sum_{k=KB}^{KC}\Delta_k^c=1-z_{kb-1}^g=1-z_b=\frac{1}{\lambda} \tag{36}$$

where *KB* equals *KC*-*KL*+1, gives

$$\partial_t\left(m_x m_y H\right)+\partial_x\left(m_y H \lambda \sum_{k=KB}^{KC}\left(\Delta_k^c u_k\right)\right)+\partial_y\left(m_x H \lambda \sum_{k=KB}^{KC}\left(\Delta_k^c v_k\right)\right)=\sum_{k=KB}^{KC}Q_k \tag{37}$$

the depth average or external barotropic mode continuity equation. Writing (35) as

$$\begin{aligned}&\partial_t\left(m_x m_y H\right)+\frac{1}{\lambda\Delta_k^c}\left(\partial_x\left(m_y \lambda H \Delta_k^c u_k\right)+\partial_y\left(m_x \lambda H \Delta_k^c v_k\right)\right)\\ &+\frac{1}{\lambda\Delta_k^c}\left(\left(m_x m_y w\right)_{z=z_k^g}-\left(m_x m_y w\right)_{z=z_{k-1}^g}\right)=\frac{1}{\lambda\Delta_k^c}Q_k\end{aligned} \tag{38}$$

and subtracting (37) gives

$$\begin{aligned}
&\partial_x\left(m_y\lambda H\Delta_k^c u_k\right)+\partial_y\left(m_x\lambda H\Delta_k^c v_k\right)\\
&-\lambda\Delta_k^c\partial_x\left(m_y H\lambda\sum_{k=KB}^{KC}\left(\Delta_k^c u_k\right)\right)-\lambda\Delta_k^c\partial_y\left(m_x H\lambda\sum_{k=KB}^{KC}\left(\Delta_k^c v_k\right)\right)\\
&+\left(\left(m_x m_y w\right)_{z=z_k^g}-\left(m_x m_y w\right)_{z=z_{k-1}^g}\right)=Q_k-\lambda\Delta_k^c\sum_{k=KB}^{KC}Q_k
\end{aligned}\tag{39}$$

the internal mode continuity equation. In the EFDC model, the internal mode equation is used to diagnostically determine the vertical velocity

$$\begin{aligned}
&\left(m_x m_y w\right)_{z=z_k^g}=\left(m_x m_y w\right)_{z=z_{k-1}^g}+Q_k-\lambda\Delta_k^c\sum_{k=kb}^{kc}Q_k\\
&+\lambda\Delta_k^c\partial_x\left(m_y\lambda H\sum_{k=kb}^{kc}\left(\Delta_k^c u_k\right)\right)+\lambda\Delta_k^c\partial_y\left(m_x\lambda H\sum_{k=kb}^{kc}\left(\Delta_k^c v_k\right)\right)\\
&-\Delta_k^c\partial_x\left(m_y\lambda H u_k\right)-\Delta_k^c\partial_y\left(m_x\lambda H v_k\right)
\end{aligned}\tag{40}$$

Since

$$w_{z=z_{k-1}^g}=w_{z=z_b}=0\quad;\quad k=KB\tag{41}$$

the recursion defined by (40) is readily solved starting from the bottom layer.

### 3.2 Hydrostatic Pressure Partitioning

Before proceeding to the transformation of the horizontal momentum equations, it is useful to first transform the hydrostatic vertical pressure partitioning in EFDC. The momentum equation in the physical $Z$ coordinate is

$$\frac{DW}{DT}=-\frac{\partial P}{\partial Z}-g\frac{\rho}{\rho_o}\tag{42}$$

where $P$ is the total kinematic pressure and $\rho_o$ is the reference density. $P$ is partitioned into non-hydrostatic, constant density hydrostatic, and variable density hydrostatic components by

$$P=p_n+g\left(Z_s-Z\right)+p\tag{43}$$

Differentiating (43) and introducing into (42) gives

$$\frac{DW}{DT}=-\frac{\partial p_n}{\partial Z}-\frac{\partial p}{\partial Z}+g\frac{\rho_o}{\rho_o}-g\frac{\rho}{\rho_o}\tag{44}$$

which can be further partitioned into a non-hydrostatic dynamic relationship

$$\frac{DW}{DT} = -\frac{\partial p_n}{\partial Z} \tag{45}$$

and a hydrostatic relationship

$$\frac{\partial p}{\partial Z} = -g\frac{\rho - \rho_o}{\rho_o} = -gb \tag{46}$$

where $b$ is the buoyancy. The current hydrostatic formulation in EFDC uses only (46) and the hydrostatic approximation of (43)

$$P \approx p + g\left(Z_s - Z\right) \tag{47}$$

The LCL transformations of (46) is

$$\frac{\partial p}{\partial z} = -g\lambda H\frac{\rho - \rho_o}{\rho_o} = -g\lambda Hb \tag{48}$$

The untransformed, $X$, horizontal gradient of (47) is

$$\frac{\partial P}{\partial X} = \frac{\partial p}{\partial X} + g\frac{\partial Z_s}{\partial X} \tag{49}$$

Transforming (49) using (28) gives

$$\begin{aligned}\lambda H\frac{\partial P}{\partial X} &= +g\lambda H\partial_x Z_s + \partial_x\left(\lambda Hp\right)\\ &-\partial_z\left(\left(\partial_x Z_s - (1-z)\partial_x\left(\lambda H\right)\right)p\right)\end{aligned} \tag{50}$$

The variable density hydrostatic pressure in (50) is determined using

$$p = g\lambda H\int_z^1 b\,dz \tag{51}$$

the integral of (48).

### 3.3 Horizontal Momentum Equations

The transformation of the horizontal momentum equations is illustrated by beginning with the X momentum equation in the physical vertical coordinate

$$\frac{\partial\left(m_x m_y u\right)}{\partial T}+\frac{\partial\left(m_y uu\right)}{\partial X}+\frac{\partial\left(m_x vu\right)}{\partial Y}+\frac{\partial\left(m_x m_y Wu\right)}{\partial Z}-m_x m_y fv$$
$$=-m_y\frac{\partial P}{\partial X}+\frac{\partial}{\partial Z}\left(m_x m_y A_v\frac{\partial u}{\partial Z}\right) \tag{52}$$

where $f$ is the Coriolis and curvature acceleration parameter and $A_v$ is the kinematic vertical turbulent diffusion coefficient. Multiplying (52) by $\lambda H$ and making use of (31) and (50) gives

$$\partial_t\left(m_x m_y\lambda Hu\right)+\partial_x\left(m_y\lambda Huu\right)+\partial_y\left(m_x\lambda Hvu\right)+\partial_z\left(m_x m_y wu\right)$$
$$-m_x m_y f\lambda Hv=-m_y g\lambda H\partial_x Z_s-m_y\partial_x\left(\lambda Hp\right) \tag{53}$$
$$+m_y\partial_z\left(\left(\partial_x Z_s-\left(1-z\right)\partial_x\left(\lambda H\right)\right)p\right)+\partial_z\left(m_x m_y\frac{A_v}{\lambda H}\partial_z u\right)$$

which can be simplified to

$$\partial_t\left(m_x m_y\lambda Hu\right)+\partial_x\left(m_y\lambda Huu\right)+\partial_y\left(m_x\lambda Hvu\right)+\partial_z\left(m_x m_y wu\right)$$
$$-m_x m_y f\lambda Hv=-m_y\lambda H\partial_x\left(gZ_s\right)-m_y\lambda H\partial_x p \tag{54}$$
$$+m_y\left(\partial_x Z_s-\left(1-z\right)\partial_x\left(\lambda H\right)\right)\partial_z p+\partial_z\left(m_x m_y\frac{A_v}{\lambda H}\partial_z u\right)$$

Introducing (48) and (51), equation (54) becomes

$$\partial_t\left(m_x m_y\lambda Hu\right)+\partial_x\left(m_y\lambda Huu\right)+\partial_y\left(m_x\lambda Hvu\right)+\partial_z\left(m_x m_y wu\right)$$
$$-m_x m_y f\lambda Hv=-m_y\lambda H\partial_x\left(gZ_s\right)-m_y\lambda H\partial_x\left(g\lambda H\int_z^1 b dz\right) \tag{55}$$
$$-gm_y\lambda H\left(\partial_x Z_s-\left(1-z\right)\partial_x\left(\lambda H\right)\right)b+\partial_z\left(m_x m_y\frac{A_v}{\lambda H}\partial_z u\right)$$

The corresponding y momentum equation is

$$\partial_t\left(m_x m_y\lambda Hv\right)+\partial_x\left(m_y\lambda Huv\right)+\partial_y\left(m_x\lambda Hvv\right)+\partial_z\left(m_x m_y wv\right)$$
$$+m_x m_y f\lambda Hu=-m_x\lambda H\partial_y\left(gZ_s\right)-m_x\lambda H\partial_y\left(g\lambda H\int_z^1 b dz\right) \tag{56}$$
$$-gm_x\lambda H\left(\partial_y Z_s-\left(1-z\right)\partial_y\left(\lambda H\right)\right)b+\partial_z\left(m_x m_y\frac{A_v}{\lambda H}\partial_z v\right)$$

For the staggered C grid used in EFDC, the depth must be defined at *u* and *v* cell faces in addition to at the cell centers. This is accomplished using

$$\begin{aligned}(\lambda H)^u &= \frac{1}{2}\frac{\left(m_x m_y\right)^w (\lambda H)^w + \left(m_x m_y\right)^e (\lambda H)^e}{\left(m_x m_y\right)^u} \\ (\lambda H)^v &= \frac{1}{2}\frac{\left(m_x m_y\right)^s (\lambda H)^s + \left(m_x m_y\right)^n (\lambda H)^n}{\left(m_x m_y\right)^v}\end{aligned} \tag{57}$$

where *e*,*w*,*s*, and *n* denote cell centers to the computational east, west, south and north of the corresponding cell faces. The scale factors on the *u* and *v* cell faces are defined by

$$\begin{aligned}\lambda^u &= \max\left(\lambda^w, \lambda^e\right) \\ \lambda^v &= \max\left(\lambda^s, \lambda^n\right)\end{aligned} \tag{58}$$

The form (57) is chosen for its energy conservation properties.

### 3.4 Transport Equation

The generic transport equation in LCL sigma vertical coordinates is obtained by adding volume source/sink and vertical diffusion terms to (31) to give

$$\begin{aligned}&\partial_t\left(m_x m_y \lambda H C\right) + \partial_x\left(m_y \lambda H u C\right) + \partial_y\left(m_x \lambda H v C\right) + \partial_z\left(m_x m_y w C\right) \\ &= \partial_z\left(m_x m_y \frac{A_b}{\lambda H} C\right) + \max\left(Q, 0\right) C_{in} + \min\left(Q, 0\right) C + m_x m_y \lambda H r_C\end{aligned} \tag{59}$$

Where $C_i$ is the inflow or source concentration, and $r_c$ is the reactive source/sink having units of mass per unit volume, unit time.

# 4. Configuration of the Generalized Vertical Coordinate in EFDC

The configuration of the generalized vertical coordinate option in EFDC is described in this section, which include example templates input files and a discussion of options for graphical post-processing.

## 4.1 Configuration Control in File EFDC.INP

The general vertical grid configuration is controlled in the EFDC.INP file, where it is activated using the IGRIDV parameter in data block 1A of the EFDC.INP file as shown below

```
C1A GRID CONFIGURATION AND TIME INTEGRATION MODE SELECTION
C
C   IGRIDH:  0 SINGLE HORIZONTAL GRID WITHOUT HORIZONTAL PARALLELIZATION
C            1 SINGLE HORIZONTAL GRID WITH HORIZONTAL PARALLELIZATION
C              GE.2, NUMBER OF HORIZONTAL GRIDS WITH HORIZONTAL DOMAIN
C              DECOMPOSITION PARALLELIZATION
C           -1 ONE DIMENSIONAL CHANNEL NETWORK WITH HEC TYPE CROSS SECTIONS
C   INESTH:  1 NO NESTING FOR IGRIDH.GE.2
C            2 FOR 2 TO 1 NESTING (FINE TO COARSE) FOR IGRIDH.GE.2
C            3 FOR 3 TO 1 NESTING (FINE TO COARSE) FOR IGRIDH.GE.2
C   IGRIDV:  0 STANDARD SIGMA VERTICAL GRID OR SINGLE LAYER DEPTH AVERAGE
C            1 GENERAL VERTICAL GRID WITH SIGMA AND LCL SIGMA REGIONS
C   ITIMSOL: 0 THREE TIME LEVEL INTEGRATION
C              1 TWO TIME LEVEL INTEGRATION
C
C1A  IGRIDH  INESTH  IGRIDV  ITIMSOL   IDUMMY
       0       0       1       0          0
```

Parameters defining the number of layers and the initialization of the vertical layering are specified in data block 9A, shown below.

```
C9A  VERTICAL SPACE-RELATED PARAMETERS
C
C       KC:  NUMBER OF VERTICAL LAYERS
C     KSIG:  NUMBER OF VERTICAL LAYERS IN SIGMA REGION FOR IGRIDV = 1
C  ISETGVC:  0 READ BOTTOM LAYER ID FROM GVCLAYER.INP
C            1 AUTOMATICALLY SET BOTTOM LAYER ID USING SELVREF, SELVREF
C              AND BELV (IN DXDY.INP) AND WRITE RESULTS TO GVCLAYER.OUT
C  SELVREF:  REFERENCE SURFACE ELEVATION IN RESCALED HEIGHT REGION (METERS)
C  BELVMIN:  REFERENCE (MINIMUM) BOTTOM ELEVATION IN RESCALED HEIGHT REGION
C  ISGVCCK:  0 NORMAL SETTING (OPTION 1 USED FOR DEBUGGING SIGMA/GVC COMPARE)
C            1 USE MULTI-LAYER BOTTOM FRICTION FOR SINGLE LAYER SIGMA
C
C9A  KC  KSIG  ISETGVC  SELVREF  BELVREF  ISGVCCK
     10  0     1        0.0      -18.0      0
```

When IGRIDV is set to zero, only the number of vertical layers, KC is read and used. When IGRIDV is set to one, KC defines the maximum number of vertical layers. For applications where the horizontal model domain includes for LCL sigma and sigma regions, KSIG defines the number of sigma vertical layers in the sigma regions. When the LCL sigma vertical coordinate is used over the entire horizontal model domain, KSIG is set to zero. When ISETGVC is zero, the local number of vertical layers is read from file GVCLAY.INP. Setting ISEGVC to one automatically initializes the vertical layering using the input values of SELVREF and BELMIN and the initial bottom elevations in DXDY.INP and writes the layer information, including the rounded bottom elevations to GVCLAY.OUT. At this point, the user has the option of allowing the model execution to proceed or updating the DXDY.INP file with rounded bottom elevations. The ISGVCCK should normally be set to zero. The dimensionless layer thicknesses for sigma, LCL sigma and hybrid applications are specified in data block 10.

```
C10 LAYER THICKNESS IN VERTICAL
C
C          K:  LAYER NUMBER, K=1,KC
C        DZC:  DIMENSIONLESS LAYER THICKNESS (THICKNESSES MUST SUM TO 1.0)
C              FOR IGRIDV=1, THE TOP KSIG LAYERS ARE PRESENT IN BOTH THE
C              SIGMA AND RESCALED HEIGHT REGIONS
C
C10     K          DIMESIONLESS LAYER THICKNESSS
        1          0.1
        2          0.1
        3          0.1
        4          0.1
        5          0.1
        6          0.1
        7          0.1
        8          0.1
        9          0.1
       10          0.1
```

## 4.2 File CELLGVC.INP

When IGRIDV is set to one, the input file CELLGVC.INP is required. The file shown below follows the format of the CELL.INP file also shown.

```
C cellgvc.inp file, i columns and j rows
C             1         2         3         4         5
C    123456789012345678901234567890123456789012345678901234567890
C
  3  000000000000000000
  2  011111111111111110
  1  000000000000000000
C             1         2         3         4         5
C    123456789012345678901234567890123456789012345678901234567890
```

```
C cell.inp file, i columns and j rows
C             1         2         3         4         5
C    123456789012345678901234567890123456789012345678901234567890
C
  3  999999999999999999
  2  955555555555555559
  1  999999999999999999
C             1         2         3         4         5
C    123456789012345678901234567890123456789012345678901234567890
```

In the CELLGVC.INP file, nonzero entries are made at I,J locations corresponding to water cells, 1-5, in the CELL.INP file. Values of one denote cells have LCL sigma vertical coordinates and values of 2 denote cells having sigma vertical coordinates when hybrid sub-domains are used.

### 4.3 File GVCLAYER.INP

When ISETGVC is set to zero, the input file GVCLAYER.INP is required and shown below for a grid corresponding to the CELL.INP file of the previous section. The 16 water cell grid shown below follows the format of the CELL.INP file also shown.

```
C  file gvclayer.inp
C  I,J = cell indices
C  KL = local number of active vertical layers
c    I     J     KL
     2     2     3
     3     2     4
     4     2     5
     5     2     6
     6     2     7
     7     2     8
     8     2     9
     9     2     10
    10     2     10
    11     2     9
    12     2     8
    13     2     7
    14     2     6
    15     2     5
    16     2     4
    17     2     3
```

## 4.4 Example of Initialization

An example of the initialization of the LCL sigma vertical grid is presented in this section. Figure 1 shows the continuous bed elevation and initial water surface elevation profiles for a 16 kilometer long closed basin with a still water maximum depth of 18 meters. The bed elevation profile is based on a full cosine wave, while the surface elevation profile is base on a half cosine wave. Figure 2 shows the same profiles with discrete bottom and water surface elevations corresponding to the DXDY.INP file shown below.

```
C dxdy.inp file, in free format across columns
C
C     I     J     DX         DY         DEPTH    BOT ELEV     ZROUGH   VEG TYPE
C
      2     2    1000.00    1000.00      8.21      -6.23       0.00      1
      3     2    1000.00    1000.00      9.01      -7.11       0.00      1
      4     2    1000.00    1000.00     10.49      -8.73       0.00      1
      5     2    1000.00    1000.00     12.39     -10.85       0.00      1
      6     2    1000.00    1000.00     14.41     -13.15       0.00      1
      7     2    1000.00    1000.00     16.21     -15.27       0.00      1
      8     2    1000.00    1000.00     17.47     -16.89       0.00      1
      9     2    1000.00    1000.00     17.97     -17.77       0.00      1
     10     2    1000.00    1000.00     17.58     -17.77       0.00      1
     11     2    1000.00    1000.00     16.32     -16.89       0.00      1
     12     2    1000.00    1000.00     14.33     -15.27       0.00      1
     13     2    1000.00    1000.00     11.89     -13.15       0.00      1
     14     2    1000.00    1000.00      9.31     -10.85       0.00      1
     15     2    1000.00    1000.00      6.98      -8.73       0.00      1
     16     2    1000.00    1000.00      5.20      -7.11       0.00      1
     17     2    1000.00    1000.00      4.25      -6.23       0.00      1
```

Running EFDC with the above DXDY.INP file for a maximum of 10 LCL sigma layers with the ISEGVC option equal to one, SELREF equal to 0.0, and BELMIN equal to -18, produces the following GVCLAYER.OUT:

```
c     I     J     KL     KB     LAMBA     BELV DXDY   BELV ROUNDED
      2     2      3      8     3.3333     -6.2300     -5.4000
      3     2      4      7     2.5000     -7.1100     -7.2000
      4     2      5      6     2.0000     -8.7300     -9.0000
      5     2      6      5     1.6667    -10.8500    -10.8000
      6     2      7      4     1.4286    -13.1500    -12.6000
      7     2      8      3     1.2500    -15.2700    -14.4000
      8     2      9      2     1.1111    -16.8900    -16.2000
      9     2     10      1     1.0000    -17.7700    -18.0000
     10     2     10      1     1.0000    -17.7700    -18.0000
     11     2      9      2     1.1111    -16.8900    -16.2000
     12     2      8      3     1.2500    -15.2700    -14.4000
     13     2      7      4     1.4286    -13.1500    -12.6000
     14     2      6      5     1.6667    -10.8500    -10.8000
     15     2      5      6     2.0000     -8.7300     -9.0000
     16     2      4      7     2.5000     -7.1100     -7.2000
     17     2      3      8     3.3333     -6.2300     -5.4000
```

where KL is the number of active layers and KB is the index for the bottom layer. The header line has been added after the file was created. To run with ISETGVC equal zero, the first three columns of GVCLAYER.OUT can be used to create the GVCLAYER.INP file shown in the preceding section. The user also has the option of replacing the original bed elevations in DXDY.INP with the rounded bed elevations output in CVCLAYER.OUT. If this is done, the initial water depths in DXDY.INP should also be updated

$$H_{rounded} = H + Z_b - Z_{b,rounded} \tag{60}$$

such that the initial water surface elevation remains unchanged.

## 4.4 Plotting Vertical Grids

Plotting vertical plane grid sections and model output variables is slightly problematic and thus a number of alternatives are available. The problematic nature arises due to the fact that the water surface and bed elevations are only formally defined and computationally used at horizontal cell centers. Water column depth is defined at the horizontal cell centers and horizontal cell faces by (57). Visually appealing plots of the model grid and cell centered variables in the vertical plane generally requires definition of either the water surface or bed elevation and possibly both on cell faces. In this section a number of alternative plotting options are discussed and illustrated graphically using the example from the preceding section. The choice as to whether to round initial bottom elevations is also discussed.

The question of how to define water surface and bed elevations on cell faces is avoided by plotting vertically plane grid sections in cell block format. Block format essentially represents each cell by assuming that the water surface and bed elevations are piecewise continuous between horizontal cell stacks. Figure 3 shows the initial block formatted vertical grid section corresponding to the non-rounded discrete bed elevations in Figure 2, with the continuous bed and water surface elevation profiles from Figure 1 also shown. Figure 4 shows a similar plot based on replacing the bottom elevations in DXDY.INP with rounded values while maintaining the initial water surface elevation. Both figures indicate that piecewise approximate of the water surface elevation is quite accurate. The piecewise approximation of the bottom elevation can be interpreted as extending the laterally constraining cell center bottom elevations to vertex corners have both 90 and 270 degree interior angles. As would be expected, the non-rounded bottom elevations, Figure 3, provide a better representation of the continuous bathymetry than the rounded bottom elevations, Figure 4. These two figures are visually unappealing in that they fail to show connectivity between the layers with interior grid vertices having multiple locations. The connectivity failure is more exaggerated for the non-rounded bottom elevations in Figure 3.

The next alternative resolves the connectivity issue and is based on retaining the piecewise bed elevations and using cell face depths, equation (57) to define the water surface elevation on cell faces. This alternative might be viewed as consistent with

model computational dynamics. Figures 5 and 6 show the grids for the non-rounded and rounded bed elevations, respective, plotted in this manner. The non-rounded bottom elevation case produces a grid plot that is not particularly appealing in that there are large errors in the water surface elevation location as well as a lack of smoothness along layer interfaces. The rounded bottom elevation plot, Figure 6, is much more appealing, with smoother layer interface lines and a smaller water surface elevation error. The smoother is better rule of grid generation could be invoke here in favor of using rounded bottom elevations.

One final alternative will be considered, although a number of additional alternative remain to be explored. This alternative uses linear interpolation of cell center water surface elevations combined in the previously used piecewise specification of bed elevations. This alternative does also have a dynamic computational basis in that the water surface elevation gradient in the momentum equations corresponds to a linear variation in water surface elevation between cell centers. Figures 7 and 8 show the grids for the non-rounded and rounded bed elevations, respective, plotted in this manner. The correspondence with the continuous water surface is good in both figures as would be expected. For the non-rounded bed elevations, Figure 7, the smoothness of layer interface lines is somewhat improved but remains poor relative to the rounded bed elevations, Figure 8.

The observations form the preceding two plotting alternatives tend to indicate that rounded bottom elevations, which invoke a uniformity of the lateral constrain concept over the entire LCL sigma model domain, are to be preferred computationally. Further model application testing is recommended to confirm this.

## 5. References

Adcroft, A. and J.M. Campin, 2004: Rescaled height coordinates for accurate representation of free-surface flows in ocean circulation models. *Ocean Modelling*, **7**, 269-284.

Ezer, T. and G. L. Mellor, 2004: A generalized coordinate ocean model and a comparison of the bottom boundary layer dynamics in terrain-following and in z-level grids. *Ocean Modelling*, **6**, 379-403.

Martin, P. J., 2000: A description of the Navy Coastal Ocean Model version 1.0. Naval Research Laboratory Report NRL/FR?7322-00-9962. Naval Research Laboratory, Stennis Space Center, MS, 42 pp.

Mellor, G. L., T. Ezer and L.-Y. Oey, 1994: The pressure gradient conundrum of sigma coordinate ocean models. *J. Atmos. and Oceanic Tech*., **11**, 1126-1134.

Mellor, G. L., L.-Y. Oey and T. Ezer, 1998: Sigma coordinate pressure gradient errors and the seamount problem, *J. Atmos. Ocean. Tech*., **15**, 1122-1131.

Stacey, M. W., S. Pond, and Z. P. Nowak, 1995: A numerical model of circulation in Knight Inlet, British Columbia, Canada. *J. Phys. Oceanogr.* **25**, 1037-1062.

# 6. Figures

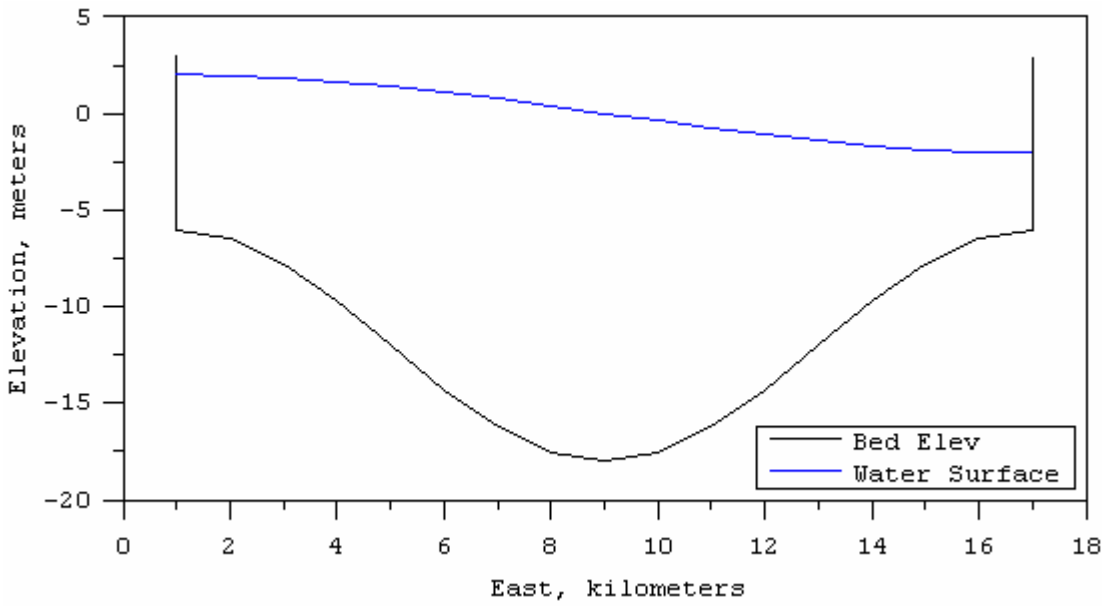


Figure 1. Example basin showing continuous bed and initial water surface elevations.

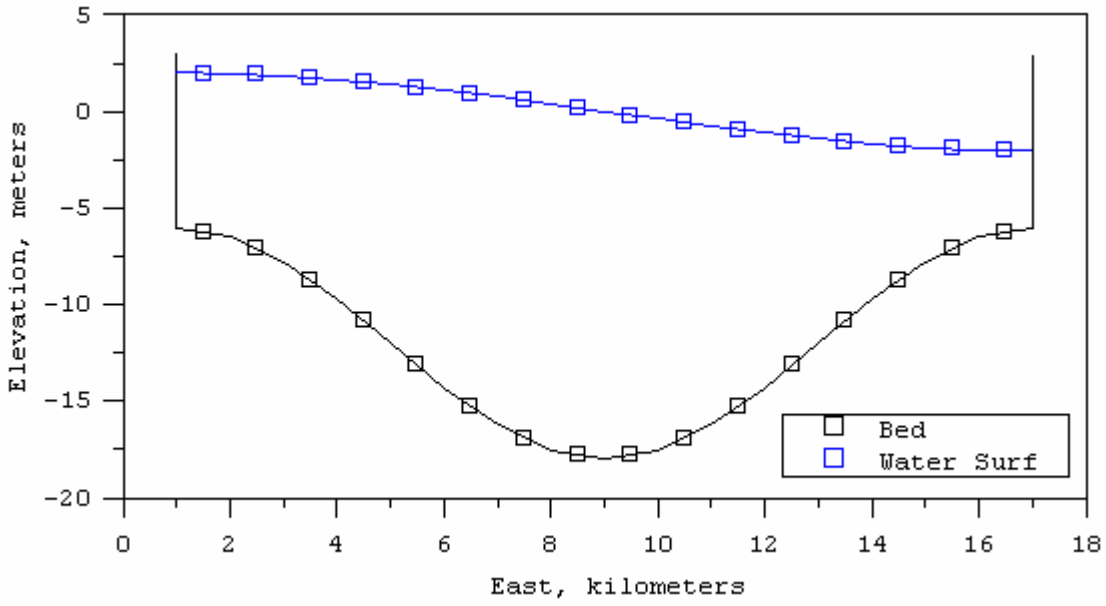


Figure 2. Example basin showing discrete bed and initial water surface elevations.

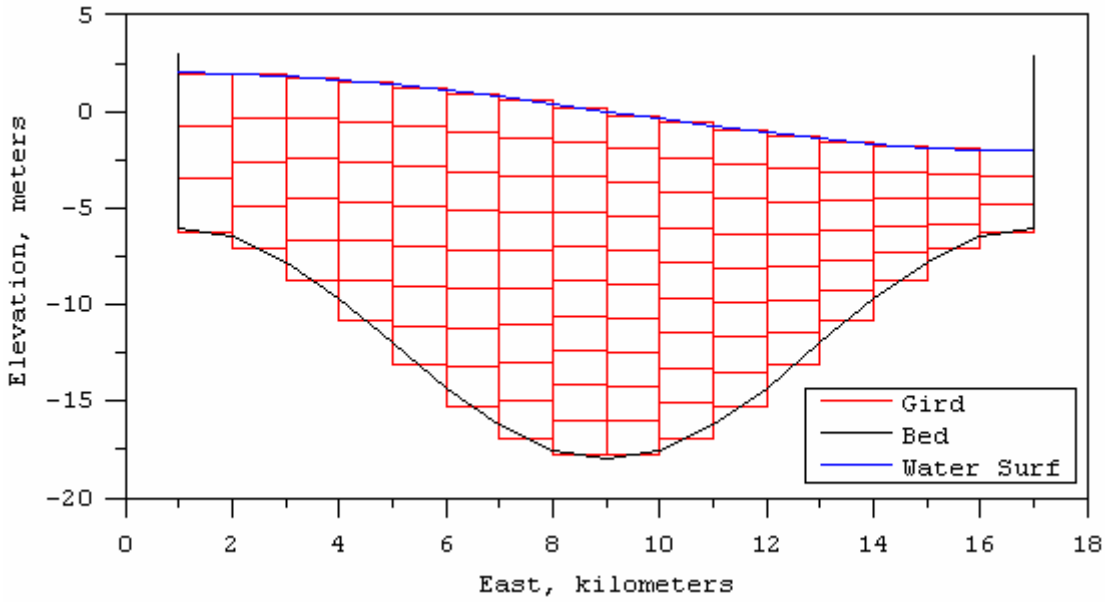


Figure 3. Example grid for non-rounded cell center bottom elevations plotted in block format.

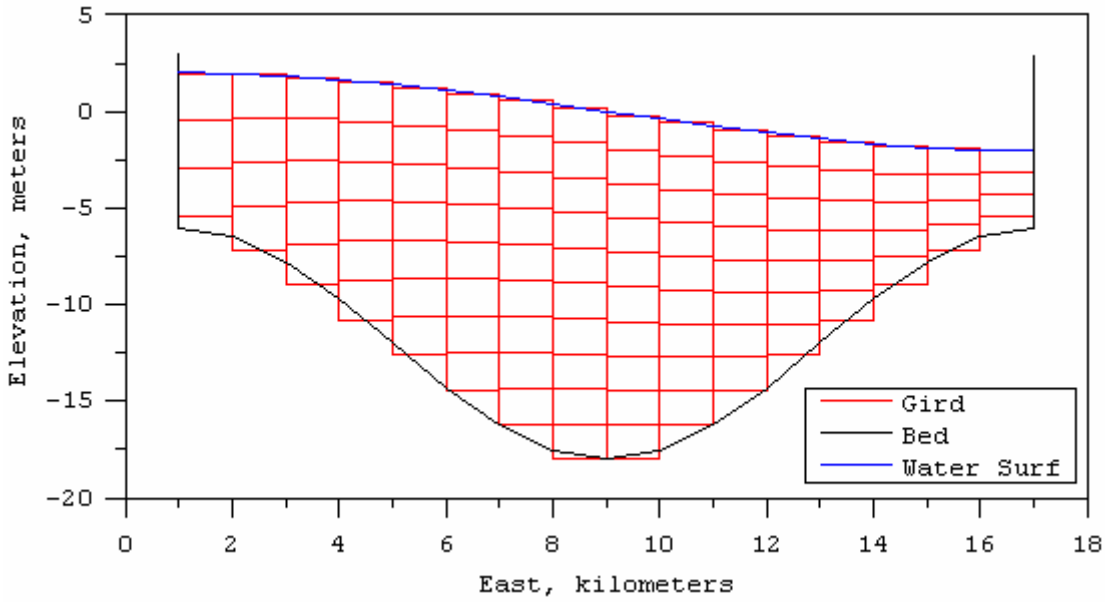


Figure 4. Example grid for rounded cell center bottom elevations plotted in block format.

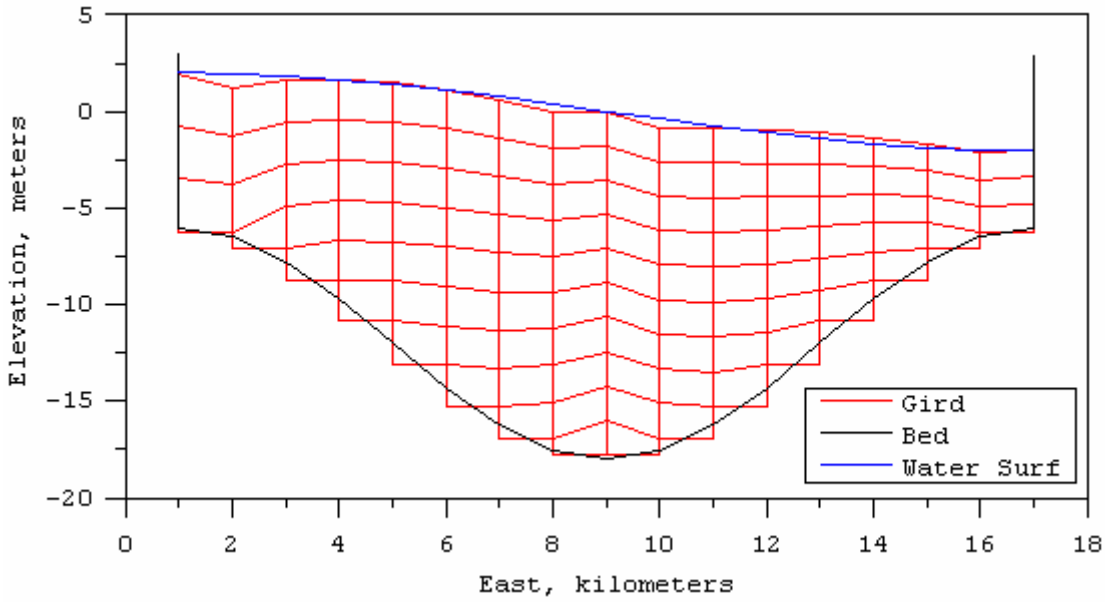


Figure 5. Example grid for non-rounded cell center bottom elevations plotted in stepped bottom and cell face depth format.

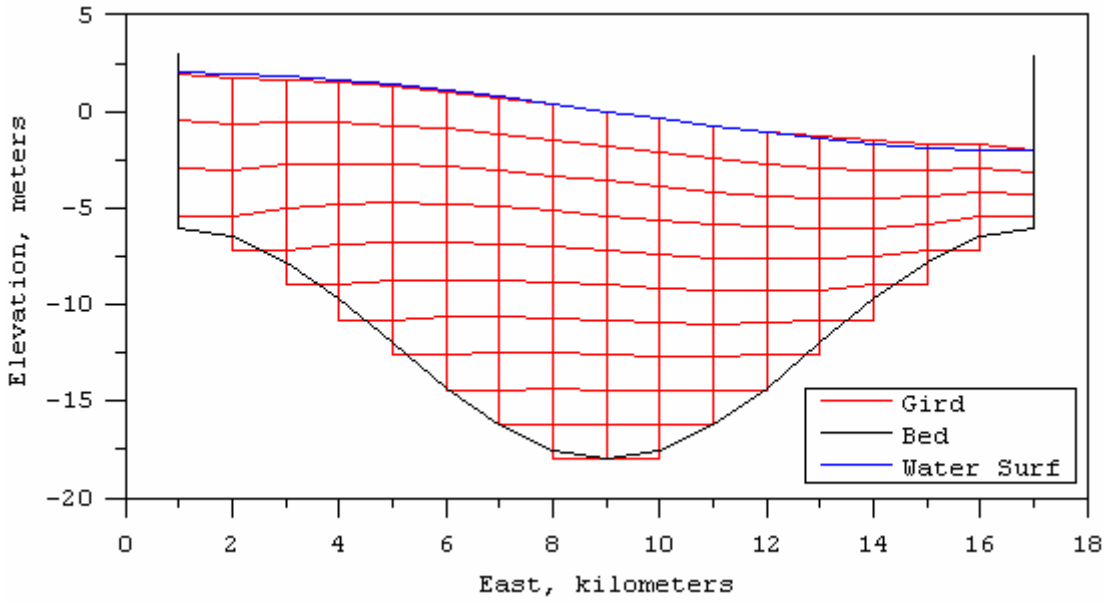


Figure 6. Example grid for rounded cell center bottom elevations plotted in stepped bottom and cell face depth format.

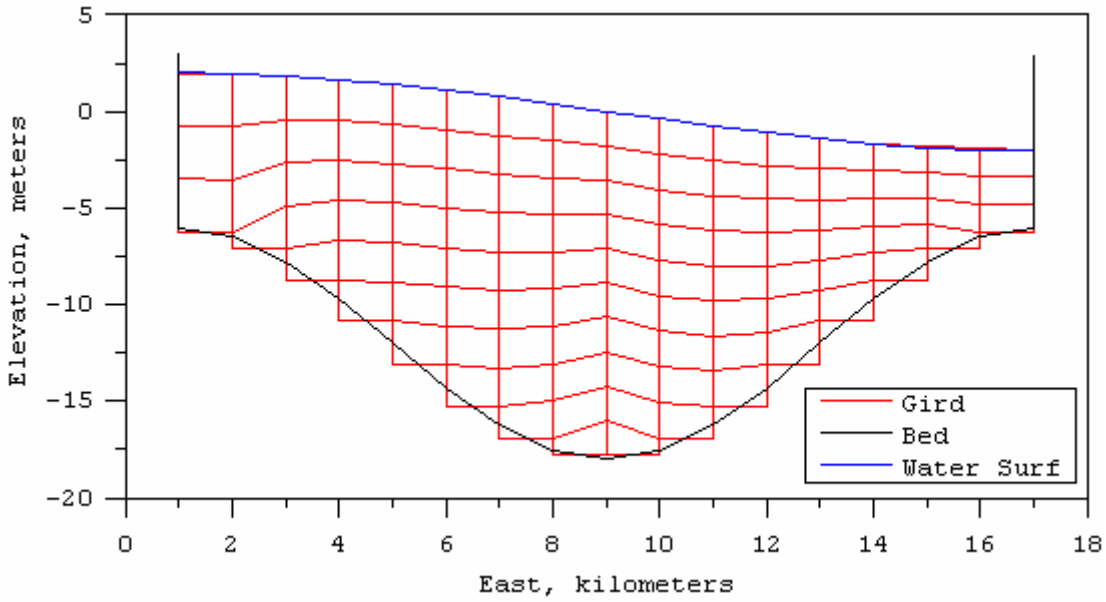


Figure 7. Example grid for non-rounded cell center bottom elevations plotted in stepped bottom and interpolated water surface format.

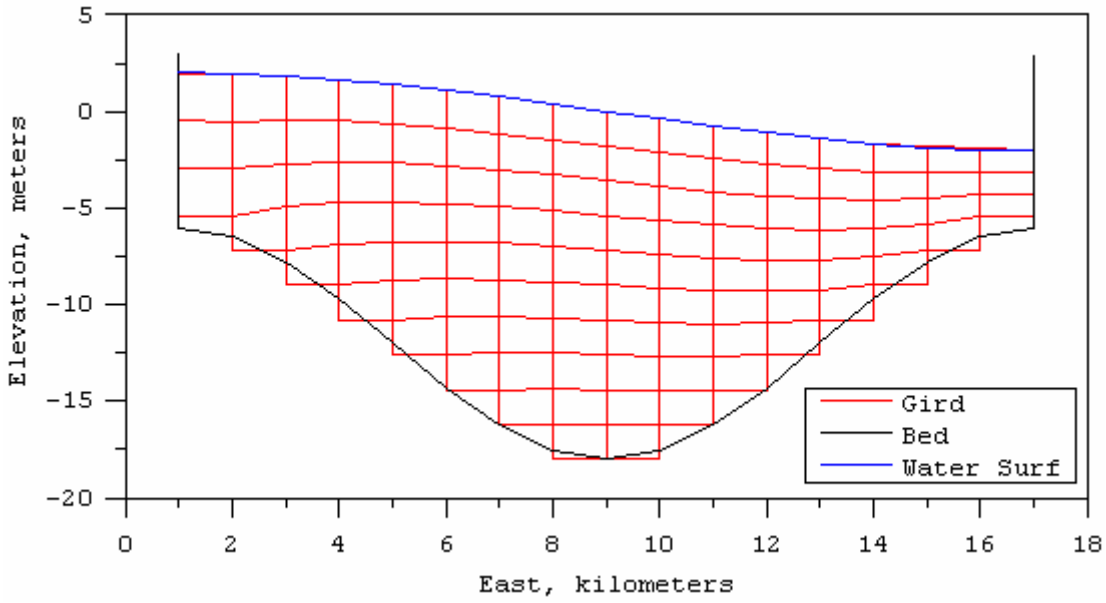


Figure 8. Example grid for rounded cell center bottom elevations plotted in stepped bottom and interpolated water surface format.